

【特許請求の範囲】

【請求項1】 各データブロックが該ブロックに関するデータを含むヘッダと、各スロットが該スロットに関するスロット・ヘッダ及び1つのデータ・パケットを有する複数のスロットとを含むデータブロックを具えるデジタル信号であって、上記データ・パケットは或るソースからの情報の次々に連続する各部分を含み、上記ソースからの情報の先頭部分を含む最初のパケットを含む最初のスロットはまた基準タイムをも含み、上記ソースからの情報の後続パケットを含む1つの又は各後続スロットはまた当該パケットの上記基準タイムに対するタイミングを定めるタイミング情報をも含む上記デジタル信号を符号化するエンコーダであって、

クロックと、

該クロックから上記最初のパケットの生成タイムを決める基準タイムを取出して、上記最初のスロット内の基準タイム情報を供給し、上記クロックから後続パケットの生成タイムを決める上記タイミング情報を取出して、上記後続パケットが生成されるに従い上記後続スロット内の上記タイミング情報を供給する手段とを具えたエンコーダ。

【請求項2】 上記タイミング情報は、粗いタイミング情報と微細なタイミング情報とを含む請求項1のエンコーダ。

【請求項3】 上記クロックは、クロック信号を受けるための入力と、該クロック信号をカウントし該クロック信号の周波数を n で割算して微細なタイム情報を生成するモジュロ n カウンタと、該モジュロ n カウンタによって生成され周波数分割されたクロック信号をカウントし粗いタイム情報を生成するモジュロ m カウンタ（ただし、 m は n よりはるかに大きい。）とを具えた請求項2のエンコーダ。

【請求項4】 上記クロック信号の周波数は $2.25n$ MHz（ただし、 n は整数）である請求項3のエンコーダ。

【請求項5】 上記クロック信号の周波数は 27 MHzであり、 n は 12 、 m は 65536 である請求項4のエンコーダ。

【請求項6】 上記クロック信号の周波数は 36 MHzであり、 n は 16 、 m は 65536 である請求項4のエンコーダ。

【請求項7】 上記タイム情報を上記スロットに挿入するマルチプレクサを具えた請求項1のエンコーダ。

【請求項8】 上記スロット・ヘッダに該スロットが上記最初のパケットを含むかどうかを示すフラグを挿入する手段を更に具えた請求項1のエンコーダ。

【請求項9】 上記データ・ブロックが可変長ブロックである請求項1のエンコーダ。

【請求項10】 上記スロットが可変長スロットである請求項9のエンコーダ。

【請求項11】 上記可変長スロットは、メタデータを含むスロットと、該メタデータによって記述されるデータを含むスロットとを具える請求項10のエンコーダ。

【請求項12】 メタデータ・スロットが、該メタデータ・スロットにおけるメタデータによって記述されるデータを含むデータ・スロットに先行する請求項11のエンコーダ。

【請求項13】 上記メタデータ・スロットは、上記最初のパケットを含む次のスロットを識別するメタデータを含む請求項12のエンコーダ。

【請求項14】 上記可変長スロットは、データ領域と、該データ領域におけるデータのタイプを記述するデータを含むタイプ領域と、上記データ領域におけるデータの長さを定める長さ領域とを具える請求項11のエンコーダ。

【請求項15】 請求項1によるエンコーダを含むSDTシステム。

【請求項16】 各データ・ブロックが該ブロックに関するデータを含むヘッダと、各スロットが該スロットに関するスロット・ヘッダ及び1つのデータ・パケットを有する複数のスロットとを含むデータ・ブロックを具えるデジタル信号であって、上記データ・パケットは或るソースからの情報の次々に連続する各部分を含み、上記ソースからの情報の先頭部分を含む最初のパケットを含む最初のスロットはまた基準タイムをも含み、上記ソースからの情報の後続パケットを含む1つの又は各後続スロットはまた当該パケットの上記基準タイムに対するタイミングを定めるタイミング情報をも含む上記デジタル信号を復号するデコーダであって、

クロックと、

上記パケットの上記タイミング情報を検出する手段と、上記最初のパケットを検出すると、上記クロックを上記基準タイムに初期設定する手段と、

上記クロックのタイムを上記後続パケットの上記タイミング情報と比較する手段と、

上記パケットの上記タイミング情報が上記クロックのタイムと等しくなる時に上記パケットを出力する手段とを具えたデコーダ。

【請求項17】 上記タイミング情報は、粗いタイミング情報と微細なタイミング情報とを含む請求項16のデコーダ。

【請求項18】 上記クロックは、クロック信号を受けるための入力と、該クロック信号をカウントし該クロック信号の周波数を n で割算して微細なタイム情報を生成するモジュロ n カウンタと、該モジュロ n カウンタによって生成され周波数分割されたクロック信号をカウントし粗いタイム情報を生成するモジュロ m カウンタ（ただし、 m は n よりはるかに大きい。）とを具えた請求項17のデコーダ。

【請求項19】 上記クロック信号の周波数は 2.25

nMHzであり、nは整数である請求項18のデコーダ。

【請求項20】 上記クロック信号の周波数は27MHzであり、nは12、mは65536である請求項19のデコーダ。

【請求項21】 上記クロック信号の周波数は36MHzであり、nは16、mは65536である請求項19のデコーダ。

【請求項22】 上記スロットが上記最初の packets を含むかどうかを示すフラグを含むスロット・ヘッダをもつ信号に使用され、上記フラグと上記 packets とを分離するデマルチプレクサと、上記フラグが最初の packets であることを示すと、該フラグに回答して上記クロックを上記基準タイムに設定する手段とを具えた請求項16のデコーダ。

【請求項23】 上記出力手段はFIFOバッファを含む請求項16のデコーダ。

【請求項24】 MPEG-2 TS packets をSDTIシステムを介して送信する送信装置。

【請求項25】 各データ・ブロックが該ブロックに関するデータを含むヘッダと、各スロットが該スロットに関するスロット・ヘッダ及びデータ・ packets を有する少なくとも1つのスロットとを含むデータ・ブロックを具えるデジタル信号であって、上記データ・ packets は或るソースからの情報の次々に連続する各部分を含み、上記ソースからの情報の先頭部分を含む最初の packets を含む最初のスロットはまた基準タイムをも含み、上記ソースからの情報の後続 packets を含む1つの又は各後続スロットはまた当該 packets の上記基準タイムに対するタイミングを定めるタイミング情報をも含む上記デジタル信号。

【請求項26】 上記タイミング情報は粗いタイミング情報と微細なタイミング情報とを含む請求項25の信号。

【請求項27】 上記の粗いタイミング情報及び微細なタイミング情報は、上記スロット・ヘッダにおける別々のワードによって表される請求項26の信号。

【請求項28】 上記スロット・ヘッダは、上記 packets が最初の packets であるかどうかを示すデータを含む請求項25の信号。

【請求項29】 上記スロット・ヘッダは、 packets ・タイプを示すデータを含む請求項25の信号。

【請求項30】 上記 packets ・タイプを示すデータは、(i) packets の長さ、(ii) packets が能動エラー訂正データを含むかどうか、の一方又は両方を示す請求項29の信号。

【請求項31】 各 packets がエラー訂正データを含む請求項30の信号。

【請求項32】 上記スロット・ヘッダに該スロットがエラー訂正データを含むかどうかを示すデータが含まれる

請求項31の信号。

【請求項33】 各スロットがエラー訂正データを含む請求項32の信号。

【請求項34】 各ブロックのスロットは、一定の長さを有し、該ブロック内で予め定めた位置を有する請求項25の信号。

【請求項35】 上記データブロックは可変長ブロックである請求項25の信号。

【請求項36】 上記スロットは可変長スロットである請求項35の信号。

【請求項37】 上記可変長スロットは、メタデータを含むスロットと、該メタデータによって記述されるデータを含むスロットとを具える請求項36の信号。

【請求項38】 メタデータ・スロットは、該メタデータ・スロット内のメタデータによって記述されるデータを含むデータ・スロットに先行する請求項37の信号。

【請求項39】 上記メタデータ・スロットは、上記最初の packets を含む後続スロットを識別するメタデータを含む請求項38の信号。

【請求項40】 上記可変長スロットは、データ領域と、該データ領域内のデータのタイプを記述するデータを含むタイプ領域と、上記データ領域内のデータの長さを定める長さ領域とを具える請求項36の信号。

【請求項41】 上記ブロック及びブロック・ヘッダはSDTIに従うものである請求項25の信号。

【請求項42】 上記 packets は、MPEG-2 TS packets 又はATMセル又はインターネット・プロトコル・ packets である請求項35の信号。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】本発明は、信号フォーマット及び信号処理装置に関するものである。詳しくいえば、本発明は、信号フォーマットと、該フォーマットによる信号を符号化する信号エンコーダと、これに対応するデコーダと、これらエンコーダ及びデコーダを含む信号送信装置とに関するものである。

【0002】

【従来の技術】例えばMPEG-2 TS (トランスポート・ストリーム) packets のような packets 化した信号を、或る位置の機器から他の機器に送信することが望まれている。MPEG-2 TS packets をDVB非同同期直列インタフェース (ASI) を介して送信することは、公知である。DVB-ASIは、或るトランスポート・ストリームを機器の特定項目の間のような2点間において送信するのに有効であるが、その他の点では比較的柔軟性がない。

【0003】

【発明が解決しようとする課題】本発明の課題は、DVB-ASIを用いるよりはるかに柔軟性のあるこの種の送信装置を提供することである。

【0004】

【課題を解決するための手段】本発明は、一面において、MPEG-2 TSパケットをSDTIシステムを介して送信する送信装置を提供するものである。このような装置は、DVB-ASIを使用するよりはるかに柔軟性がある。SDTI（シリアル・データ・トランスポート・インタフェース）は、SMPTE305Mに規定されている。SDTIは、テレビジョン走査線（ライン）のフレームを有する信号構造でパケットを送信するものである。補助データはこれらのラインの水平ブランキング領域で送られ、データは各ラインのペイロード領域で送られる。このペイロード領域は、活きている（能動）ライン間隔内にある。SDTIは、TSパケットをSDI接続が使用できるところにでも送ることができると共に、TSパケットを1より多くのソースから送信することができる。しかし、TSパケットをSDTIを介して送るには、パケットを確実にSDTIのペイロード領域内に局限すると共に、能率上各ラインが多数のパケットを含みうるようにするために、バッファリング（緩衝記憶）が必要になる。このバッファリング過程でパケットに遅延とジッタ（即ち、パケット相互のタイミングのずれ）が生じるが、MPEG-2信号を正確に復号するには、該信号のパケットを互いに正確なタイミングでMPEGデコーダに供給して正確に復号できるようにしなければならない。パケットの絶対的な遅延は、すべてのパケットに均等に影響するので問題ではないが、ジッタについてはMPEGデコーダで又はその前にこれを補正する必要がある。

【0005】上述では、MPEG-2 TSパケットをSDTIを介して送信する場合に本発明が直面する技術的問題を説明したが、他のデータ伝送システムを介する他のタイプの時間に敏感なパケット送信においても、似たような問題が起きるのであろう。

【0006】本発明は、他の面において、各データブロックが該ブロックに関するデータを含むヘッダと、各スロットが該スロットに関するスロットヘッダ及び1つのデータパケットを有する少なくとも1つのスロットとを含むデータブロックを具えるデジタル信号であって、上記データパケットは或るソースからの情報の次々に連続する各部分を含み、上記ソースからの情報の先頭部分を含む最初のパケットを含む最初のスロットはまた基準タイムをも含み、上記ソースからの情報の後続パケットを含む1つの又は各後続スロットはまた当該パケットの上記基準タイムに対するタイミングを定めるタイミング情報をも含む上記デジタル信号を提供する。

【0007】本発明は、更に他の面において、各データブロックが該ブロックに関するデータを含むヘッダと、各スロットが該スロットに関するスロットヘッダ及び1つのデータパケットを有する少なくとも1つのスロットとを含むデータブロックを具えるデジタル信号であっ

て、上記データパケットは或るソースからの情報の次々に連続する各部分を含み、上記ソースからの情報の先頭部分を含む最初のパケットを含む最初のスロットはまた基準タイムをも含み、上記ソースからの情報の後続パケットを含む1つの又は各後続スロットはまた当該パケットの上記基準タイムに対するタイミングを定めるタイミング情報をも含む上記デジタル信号を符号化するためのエンコーダを提供する。

【0008】そのエンコーダは、クロック（刻時装置）と、該クロックから上記最初のパケットの生成タイムを決める基準タイムを取出して、上記最初のスロット内の基準タイム情報を供給し、上記クロックから上記後続パケットの生成タイムを決める上記タイミング情報を取出して、上記後続パケットが生成されるに従い上記後続スロット内の上記タイミング情報を供給する手段とを具える。

【0009】本発明は、更に別の面において、各データブロックが該ブロックに関するデータを含むヘッダと、各スロットが該スロットに関するスロットヘッダ及び1つのデータパケットを有する少なくとも1つのスロットとを含むデータブロックを具えるデジタル信号であって、上記データパケットは或るソースからの情報の次々に連続する各部分を含み、上記ソースからの情報の先頭部分を含む最初のパケットを含む最初のスロットはまた基準タイムをも含み、上記ソースからの情報の後続パケットを含む1つの又は各後続スロットはまた当該パケットの上記基準タイムに対するタイミングを定めるタイミング情報をも含む上記デジタル信号を復号するためのデコーダを提供する。

【0010】そのデコーダは、クロックと、上記パケットのタイミング情報を検出する手段と、上記最初のパケットを検出すると、上記クロックを上記基準タイムに初期設定する手段と、上記クロックのタイムを上記後続パケットの上記タイミング情報と比較する手段と、該パケットの該タイミング情報が上記クロックのタイムと等しくなった時に該パケットを出力する手段とを具える。

【0011】信号内に上記タイミング情報を与えることにより、上記デコーダは、ジッタが十分に減少され望ましくは除去されたパケットを出力することができ、例えば、パケットがMPEG-2 TSパケットの場合、後続のMPEGデコーダにおいて正確な復号が可能となる。パケットのジッタは、復号を台無しにする可能性があるが、上記デコーダは、各パケットのタイミング情報を最初のパケットの基準タイムに設定された内部クロックと比較し、クロックのタイムとパケットのタイムが一致した時にパケットを、例えばバッファに出力する。これにより、ジッタは少なくとも軽減される。

【0012】本発明の一実施例では、上記ブロックは、ラインの非能動領域にアドレスデータを含む補助データをもつSDTI信号の能動ライン間隔内のペイロード領

域である。SDTIに適用される本発明の一実施形態では、SDTIライン毎に、不使用スペースに詰め物（例えばゼロ）を施した一定の整数の完全なパケットがある。即ち、スロットの長さは一定であり、個々のSDTIライン上のスロット及びパケットはすべて、同じソースと行先のアドレスを有し、そのライン上にはただ1つのパケットストリームがある。

【0013】SDTIに適用される本発明の他の実施形態では、データブロックは、1本以上のラインを占める可変長ブロックである。この可変長データブロック内のスロットは、可変長スロットである。これらのスロットは、データ領域Dにおいてデータのタイプを記述するデータを含むタイプ領域Tと、データ領域におけるデータの長さを定める長さ領域Lとを含む「TLDブロック」であるのがよい。このTLDブロックは、メタデータブロック及びデータブロックの2つのタイプがあるのがよい。メタデータTLDブロックは、該メタデータによって記述されるデータを含むデータTLDブロック（単数又は複数）に先行する。データTLDブロックは、基準タイムとタイミング情報とを含む。これらのブロックに先行するメタデータTLDブロックは、これらのブロックのどれが上記基準タイムを含むかを見分けるデータを含む。

【0014】

【発明の実施の形態】以下、図面を参照して本発明を具体的に説明する。

第1実施形態：SDTI-TS

概要

まず最初に、SDTI（SMPTE305M）を介してMPEG-2トランスポート・ストリーム（TS）パケットを送るための新しい信号フォーマットの第1の実施形態を説明する。

【0015】図1は、SDTIラインのペイロード領域の模式図である。図2のa～cは、MPEG-2 TSパケットの模式図である。便宜上、図ではデータを並列フォーマットで表しているが、実際はデータは直列フォーマットで送信される。MPEG-2 TS及びSDTIは周知であるので、ここでは詳しく説明しない。新しいフォーマットを本明細書ではSDTI-TSということにする。SDTIは、データを送るのにテレビジョンのラインを使用している。能動ライン間隔が、SDTIデータブロックを含むペイロード領域である。ラインの非能動間隔は、能動ライン間隔内のデータのソース及び行先のアドレスを含む補助データを有する。本発明の第1実施形態では、転送能率をよくするため、SDTI-TSの各ラインのペイロード領域は、1つのトランスポート・ストリームからの複数パケットを含む多数の固定長のスロットでできるだけ一杯に満たされる。新しいフォーマットは、SDTI-TSデコーダにおいて低いジッタレベルでトランスポート・ストリーム・パケットを

再生するのに必要な全情報を含んでおり、幾つかの設計要件を満たす柔軟性を有している。

【0016】参考文献

次の文献は、公表された規格であり、本発明と関係がある情報を含んでいる。

- 1) SMPTE305M, Serial Data Transport Interface (SDTI).
- 2) SMPTE-RP68, Definition of Vertical Interval Switching Point.
- 3) ISO/IEC13818-2: Information Technology-Generic Coding of Moving Pictures and Associated Audio Information: Video, (MPEG-2).
- 4) ISO/IEC13818-2: Information Technology-Generic Coding of Moving Pictures and Associated Audio Information: Systems, (MPEG-2).
- 5) DVB: Interfaces for CATV/SMATV Headends and Similar Professional Equipment; Asynchronous Serial Interface (ASI).
- 6) SMPTE259M, 10-bit 4:2:2 Component and 4fsc NTSC Composite Digital Signals/Serial Interface.
- 7) SMPTE291M, Ancillary Data Packet and Space Formatting.

【0017】背景

MPEG-2トランスポート・ストリーム（TS）パケットの送信用として現在定着しているインタフェースは、デコーダ・バッファへの影響を確実に微小とするために必要な小さい許容誤差の位置内にパケットを置くDVB-ASIである。しかし、ASIは、1つより多くのTSを容易に送ることができず、また普通に入手できるSDIを使用する機器によって支えることもできない。ASIは、機器の特定項目の間の特殊な2点間（ポイントツーポイント）インタフェースとしてのみ有用である。

【0018】本発明の図解例

本発明は、一面において、上述のSDTIを介してMPEG-2 TSパケットを送信することを提案するものである。こうすると、TSパケットをSDI接続が使用できるどこにでも送ることができるので、接続性への更に一般的な接近が可能となる。しかし、SDTIを介してのTSパケットの送信は、パケットを確実にSDTI

のペイロード領域に局限すると共に、能率上各ラインが多数のTSパケットを含みうるようにするために、バッファリングが必要になる。このバッファリング処理の結果、パケットストリームに遅延とジッタの両方が生じる。遅延は、本発明に関係する問題ではない。この明細書で述べる本発明の実施形態の例は、パケットのジッタを問題としないレベルに下げうるものである。ベンチマークとして、本実施形態は、DVB-ASI入力をSDTI-TSを介して送信し、復号して最少限のジッタしか生じない新しいDVB-ASI信号を作成することができる。上記SDTI-TSはまた、必要に応じ他のインタフェース場所から直接操作することもできる。

【0019】SDTIパラメータ

SMPTE305Mは、SDTIのビットレートを270Mbps及び360Mbpsと指定しているが、本発明の実施形態では両ビットレート及びもっと高いビットレートで作動できる。本明細書で述べるSDTI-TSの例では、SDTIのライン当たり1ブロックの固定ブロックサイズ・モードを使用する。各ブロックが固定長のフォーマットを図1に示す。このブロックは、1438又は1918バイトのどちらか一方の長さを有し、後述のような10ビットSDTIタイプ・ワードを含む。このブロックはまた、付加的な2バイトのエラー検出CRCを一般に含んでいて、図1に示すように合計で1440又は1920バイトとなる。

【0020】タイプ・ワード

SDTIデータブロックは、1つのブロック・タイプ・ワードを有する。270MbpsのSDTIの場合、そのブロックタイプ値は、ライン当たり1438ワード（該タイプ値を含む）のブロック・サイズを指定する「01h」である。360MbpsのSDTIの場合、そのブロックタイプ値は、ライン当たり1918ワード（該タイプ値を含む）のブロック・サイズを指定する「09h」である。

【0021】SDTIデータ・タイプ・ワードは、適当な値を有する。現在のところ、決められた値はない。入力フォーマットは、10ビットワードのうちのビットb7～b0の中に入る8ビットデータである。10ビットワードのうちのビットb8はビットb7～b0の偶数パリティであり、ビットb9はビットb8の補数である。

【0022】任意のFEC（フォワード・エラー・コレクション）を上記ブロックに加えてデータに安全性を加えてもよい。FECを付加するかしないに拘らず、ブロック・タイプ・ワードのビットb7及びb6は、SMPTE305Mに従って設定する。本発明のこの例では、SMPTE305Mのデータ拡張の便宜は用いない。SMPTE-RP168に規定するようなSDIスイッチング・ラインの使用は、避けた方がよい。

【0023】MPEG-2トランスポート・ストリーム
図2のaに示すように、MPEG-2 TS（トランス

ポート・ストリーム）パケットは、長さが188バイトである。図2のbに示すように、DVBのフォワード・エラー・コレクション（FEC）を付加すると、パケットの長さは204バイトになる。このMPEG-2 TS FECは、パケットの間にインタリーブされており、急に現れるかも知れない伝送エラーによる重大なデータロスを補正するためのものである。極端な場合にSDTIが偶発的なエラーを生じる可能性はあるが、MPEG-TSのFECは、そのインタリーブの長さのためSDTIエラーの訂正には不適當であり、送られるパケットがたった188バイトの長さの場合、このFECは存在しないことになる。よって、本発明の実施形態のSDTI-TSフォーマットは、任意に選択できるFECを含み、SDTIリンクに生じるかも知れないエラーを訂正するようにしている。

【0024】ペイロード・フォーマット

本発明の第1の実施形態では、MPEG-2 TSパケットは、TSペイロード・フォーマットの中のSDTI固定長ブロック内にすっぽり含まれており、以下その例を図3を参照して説明する。図示したTSペイロード・フォーマットは、図3に示すように2層構造を有する。ペイロードは、ペイロード・ライン全体に適用するパラメータを定めるTSペイロード・ヘッダと、MPEG-2 TSパケットをその中に含む一定数のTSスロットとを有する。各TSスロットは、該スロット内のTSパケットに関するパラメータを定めるスロット・ヘッダを有し、これにTSパケット自体が続き、6バイトの任意のSDTIのRS-FECで終わっている。

【0025】TSペイロード構造

TSペイロードのSDTIタイプワードは、図1の左側に示したタイプワードである。スロット・カウント、チャンネル・ハンドル及び連続カウン트의各ワードのビット割当てを図4に示す。

【0026】TSペイロードのスロット・カウント

スロット・カウントは、このペイロード・ラインにおけるTSスロットの数を定める8ビットのワードである。TSスロットの数は、下の値が1で、上の値はペイロードの長さで決まる。270MbpsのSDTIの場合、TSスロットの最大数は6である。360MbpsのSDTIの場合、TSスロットの最大数は8である。SDTIのレートが高い程、ペイロード領域により多くのTSスロットを適切に維持することができる。

【0027】本発明のこの実施形態では、最後のTSペイロード・スロットとSDTI-CRCとの間のデータスペースは、どんな他の目的や用途にも使用されない。ペイロード・スロット・カウン트의値が低い程、コードック（符号化-復号）の遅延を少なくするのに役立つ。これに関しては、あとで更に説明する。

【0028】TSペイロードのチャンネル・ハンドル
チャンネル・ハンドルは、最大有効ワードが先にある2

つの8ビットワードとして構成された16ビットのワードである。このワードは、同じSDTIのソースと行先のヘッダ・アドレスを有するものの、異なるTS送信チャンネルを表すTSペイロード・ラインを見分けるのに使用される。本例では、チャンネル・ハンドル値を「0000h」に設定するが、他の値を選んでよい。

【0029】TSペイロードの連続カウント

この「連続カウント」は、同じSDTIのソースと行先のアドレス及び同じチャンネル・ハンドル値をもつTSペイロード・ライン毎に1つずつインクリメントするモジュロ65536カウントである。このカウントの目的は、信号路のスイッチングを検出することである。

【0030】TSスロット構造

本例では、各TSスロットは216ワードの長さを有し、TSスロットは、TSペイロード・ヘッダの直ぐ後に連続して順に配列される。1つのTSスロットは、次の順序で情報を含んでいる。

2ワード長のTSスロット・ヘッダ、2.25MHzで刻時される2ワード長の粗いカウント値、2.25MHzの倍数で刻時される1ワード長の細かい(精)カウン

b1	b2	TSタイプ
0	0	16バイトFECのない188バイトのTSパケット
0	1	残してあるが未定義
1	0	非能動16バイトFECのある204バイトTSパケット
1	1	能動16バイトFECのある204バイトTSパケット

【0033】TSスロット粗カウント

粗カウント値は、SDTI-TS符号化部にある、2.25MHzで刻時されるモジュロ65536カウンタから得られ、デコーダのためのTSパケットの粗いタイミングを定めて、前のTSパケットに対し正確なタイミングでTSパケットを出力できるようにするものである。この粗いタイミングは、次のワードの、2.25MHzの整数倍のクロックから得られる微細なタイミングを含むTSスロット・サブカウントと共に使用され、TSプログラム・クロック・レファレンス(PCR)に近い基準タイミングをクロックに与えるものである。これらの粗いタイミングと細かいタイミングは、例えば27MHzで作動するPCRクロックと同じマスタークロックから得るのがよい。

【0034】2.25MHzカウンタは、PCRの27MHzの周期と調和する必要はない。2.25MHzカウンタは、TSパケット間の最大周期より十分に長いことが保証される周期を必要とするだけである。2.25MHzカウンタの周期は、約30ミリ秒である。2.25MHzカウンタは、PCRの12インクリメント毎に1つずつインクリメントするものである。

【0035】TSスロット・サブカウント

270MbpsのSDTIの場合、サブカウントは27MHzクロック(PCRと同じレートである)によって刻時され、0~11の範囲にわたってカウントされる。

ト値、188バイトのTSパケット、任意のMPEG-2 TS-FECのための16バイトのスペース、任意のSDTI RS-FECのための6バイトのスペース、2バイトのナル(ゼロ)データスペース。

【0031】TSスロット・ヘッダ

TSスロット・ヘッダは、次のように定義されたビットb7~b0より成る8ビットのワードである。ビットb7は、TSスロットが能動SDTI-FECを有するかどうかを定める。b7はSDTI-FECが能動でなければ「0」であり、SDTI-FECが能動ならば、「1」である。ビットb6~b3は、残してあるものの未定義である。ビットb2は、TS不連続フラグである。現在のTSスロット内のTSパケットが前のTSパケットと関連がない場合、b2は「1」に設定され、そうでない場合は「0」に設定される。不連続なTSパケット(即ち、b=「1」)は、これが新しいTSパケット連続列における最初のパケットであることを示す。

【0032】ビットb1及びb0は、次のようなTSパケット・タイプを定めるコードを形成する。

360MbpsのSDTIの場合、サブカウントは36MHzクロック(PCRより高いレートである)によって刻時され、0~15の範囲にわたってカウントされる。サブカウント値に使用可能なビット数は、SDIワード(並列)クロックレートを576MHzまで可能とするものである。TSスロットのカウントをどのようにして出力タイミングの制御に使用できるかについては、あとで述べる。

【0036】TSパケット・データ

TSパケットのための204バイトのスペースは、固定的に割当てられたものである。このスペースの中身は、TSスロット・ヘッダ・ワードのビットb1及びb0の定めるところに従って188バイト又は204バイトのTSパケットデータで満たされる。188バイトのTSパケットの場合及び非能動TS-FECをもつ204バイトのTSパケットの場合、16バイトのTS-FECスペースはゼロで埋められる。

【0037】TSスロットRS-FEC

このFECは、粗カウント・ワードから最後のFECワードまで214バイトのTSスロットにわたって適用される。TSスロット・ヘッダ・ワードFEC有効ビット(b7)が「0」に設定される場合、このFECの6ワードはすべて「00h」に設定される。RS-FECは、元のRS(リードソロモン)(255, 249, T=3)コードから短縮されたRS(214, 208, T

=3)コードである。

$$RS(x) = (x \oplus \alpha^0) \cdot (x \oplus \alpha^1) \cdot (x \oplus \alpha^2) \cdot (x \oplus \alpha^3) \cdot (x \oplus \alpha^4) \cdot (x \oplus \alpha^5)$$

ただし、 α はガロア体GF(256)生成多項式より定義されるものである。

$$GF(x) = x^8 \oplus x^4 \oplus x^3 \oplus x^2 \oplus 1$$

【0039】TSスロット・ナル・スペース

TSスロットは、ゼロを埋めた2ワードのナル・スペースで終わる。

【0040】TSスロット・タイミング・データの使用

図5は、MPEG-2 TSパケットストリーム入力からSDTI-TSフォーマットにより出力するまでのSDTI-TSの符号化過程を示す説明図である。TSスロット粗カウント値とサブカウント値との組合せにより、少なくともPCRと同じ位に高い分解能と、約30ミリ秒の再生成前の周期とをもつカウント値が定まる。1連続列における最初のTSパケットを受信すると、デコーダは、準備でき次第いつでもパケットを出力すると共に、SDTIワードレート(27/36MHz)で動作する内部カウンタをそのTSスロット・カウント値に設定する。このカウンタは、粗カウント及びサブカウントについて定めたと同じやり方でカウントを行う。その後、残りのすべてのパケットについて、デコーダは、デコーダのカウントがTSスロット・カウンタに記憶されたカウントと等しくなると、パケットを出力してTSパケット出力タイミングの完全さを保証する。270Mbps SDTIの場合、カウント値はTSパケットPCRと同じレートで刻時し、タイミングのジッタは生じない筈である。360Mbps SDTIの場合、カウントはTSパケットPCRの3インクリメント毎に4ずつインクリメントし、従って+又は-1のPCRクロック・ジッタが生じる。この値は、十分MPEG-2が定める限界内である。

【0041】システム遅延の問題

各SDTIラインのペイロード領域は、できるだけ多くの完全なスロット、即ちTSパケットでバックされる。本実施形態では、スロットは固定長であり、SDTIペイロード領域は完全なスロットのみを含む。ペイロード領域における予備スペースは、ゼロで埋められる。こうすれば、SDTIラインの浪費が確実に最少となり従って最大数のラインを他のデータの転送に使用できる。

【0042】次の例は、270Mbps SDTI-TSに基づくものである。4Mbpsのビットレートでは、フレーム(525/30)当たり約15ラインがMPEG-2 TSパケットの送信に使用され、コーデックの遅延は2ミリ秒をほんの少し越えるにすぎない。50Mbpsのビットレートでは、フレーム(525/30)当たり約185ラインがMPEG-2 TSパケットの送信に使用され、従ってコーデックの遅延は約200マイ

ロ秒に減少する。

【0043】まず得られた近似の結果では、コーデックの遅延はビットレートに逆比例する。よって、低いビットレートでは、遅延は、ライン当たりのTSパケット数を減らす、即ちそれに比例して一層多くのSDTIラインを占めることでしか減らすことができない。上記4Mbpsの例で仮にライン当たりたった1TSパケットを占めるようにすれば、遅延は約400マイクロ秒に減るであろう。

【0044】SDTI-TSシステムの例

図6に、SDTI-TSシステムの一例を示す。このシステムは、データソース1を有する。データソース1は、公知のMPEG-2 TSパケットを生成する公知のMPEG-2エンコーダ2を含む。このパケットは、図2のa~cに示すようなFECデータを含むこともあり、含まないこともある。このパケットは、MPEGエンコーダ2から正確な相対的タイミングで出力されるので、ジッタは殆どない。ソース1のタイミング・データ挿入回路4(図7及び8を参照してもっと詳細に述べる。)は、ヘッダに粗と微細のタイミング・データを挿入する。このタイミング・データは、パケットの基準タイムに対する正確なタイミングを定めるものである。MPEG-2 TSパケットは、タイミング・データと共にSDTI 6に送られる。SDTI 6はまた、他のソース3及び5からパケット及び他のデータをも受ける。他のソースは、ソース1と同じか又は同じでないこともある。SDTIは、ソース1、3及び5からのデータを公知のやり方で夫々対応する行先11、13及び15に送る。その目的のために、1本のSDTIラインは、その補助データ内にソース・アドレス及び行先アドレスを有し、アドレスされたソースからアドレスされた行先へのデータのみを伝送する。前述のように、MPEG-2 TSパケットはSDTI内で緩衝記憶されるので、ジッタの発生を招く。行先11では、ソース1からのパケットは、補正回路8でタイミングの補正を受けた後、正しいタイミングでMPEGデコーダ10に送られる。補正回路8の例は、図7を参照して説明する。

【0045】SDTIは、MPEG-2 TSパケットを組合せて図3に示すスロットを作成する。即ち、スロット・ヘッダ・データをタイミング・データに加え、ブロック・ヘッダ・データを同じく図3に示すようにブロックに加える。SDTIはまた、6バイトのFEC、ナルスペースに対するゼロ及び任意のSDTI-CRCを

も加える。SDTIはまた、パケットが新しいパケット連続列の先頭パケットとなる時点を決し、TS不連続フラグのビットb2を「1」にセットする。

【0046】図7に示すタイミング・データ挿入回路4の例は、カウンタ42を、刻時してタイミング・データを生成する27MHzクロック40を有する。このクロック40は、SDTIクロックから得たものである。該カウンタの例を図8に示す。該カウンタは、27MHzで刻時して微細な（精）タイミング・データを生成するモジュロ12カウンタ80を含み、これはまた、粗いタイミング・データを生成するモジュロ65536カウンタ82を刻時させる2.25MHzクロックを形成する。カウンタ80及び82は、自走型である。ラッチ44は、上記精粗のカウンタを一時的に記憶する。これは、パケット・スタート信号を受けると、現在のカウンタをマルチプレクサ(MUX)46に供給し、マルチプレクサは、パケットの先頭のビットストリームの中にタイミング・データを挿入する。1パケット連続列の先頭パケットに、その生成の瞬間にカウンタ42によって示される任意のタイムが割当てられる。そのタイムは、該連続列の後続のパケットに対する基準タイムとなる。カウンタ42は、 $65536 / (2.25 \times 10^6)$ 秒、即ち約30ミリ秒の周期でそのカウンタを続ける。

【0047】行先における補正回路8は、例えば、スロット・ヘッダを検出し、それから不連続フラグ(b2=1)及び精粗タイミング・データによって示されるパケットが連続列内の先頭パケットかどうかを決定するデマルチプレクサ(DEMUX)84を具える。デマルチプレクサは、MPEG-TSパケットをスロット・ヘッダから分離し、該パケットをFIFOバッファ96に供給する。このFIFOバッファ96は、SDTIの1ブロック分のパケットを記憶する。パケットは、平均して、ソース1で最初に生成されたときと同じレート、例えば270MbpsでFIFOバッファに刻々と入力され、それから刻々と出力される。ただし、パケットは不規則に入力されても、正確なタイミングで連続して出力される。デマルチプレクサ84は、タイミング・データをもう1つのFIFO86を介して比較回路88に送ると共に、直接ゲート回路94にも送る。パケットが最初のパケットであれば、不連続フラグb2=1によりゲート回路94が基準タイムを表すタイミング・データをカウンタ90へ供給し、これにより該カウンタが基準タイムにセットされる。

【0048】図9に示すカウンタ90の例は、図8の対応するカウンタ80及び82と同じ様に配置されSDTIによって設けられた27MHzクロックをカウントするモジュロ12カウンタ921及びモジュロ65536カウンタ941を具える。SDTIによって設けられたクロックは、公知のやり方でソースのクロック40と同期化されている。カウンタ921及び941は、予め基

準カウントを格納できる点においてカウンタ80及び82と相違する。カウンタ921及び941は、一旦格納されるとカウンタ80及び82と同様に自走し、これらと遅れて同期状態になる。

【0049】比較回路88は、デマルチプレクサからのタイミング・データをカウンタ90からのタイミング・データと比較する。比較したタイミング・データが等しくなると、比較回路は、FIFOバッファ96に正しいタイミングでパケットを出力させることができる。

【0050】パケット及びタイミング・データは、デマルチプレクサ84によって分離され、FIFO86及び96に供給される。これらのFIFOは、対応するパケットとタイミング・データの配列を一時記憶する。パケットがFIFO96を通してその出力に移動すると、対応するタイミング・データも、FIFO86を通して移動しFIFO96のパケットと対応関係を維持する。タイミング・データの各項目がFIFO86の出力に達したあと、その項目はそのFIFOから除かれる。FIFO86の出力におけるタイミング・データの各項目はカウンタ90のカウントと比較され、タイミング・データによって表されるカウントが該カウンタにおけるカウントと等しくなると、読出しイネーブル信号がFIFO96に正しいタイミングをもつ対応するパケットの読出しを開始させる。このイネーブル信号はまた、FIFO86の出力からタイミング・データを除去し、タイミング・データの次の項目をその出力に移してカウンタ90のカウントと比較させる。

【0051】第2実施形態：SDTI-PF

概要

図1～4について、本明細書でSDTI-TSという信号フォーマットを説明した。それは、ライン当たり1ブロックの、SDTIの固定長ブロック・サイズ・モードを用いるものであった。その信号フォーマットを、SDTIを使用してMPEG-2ビットストリームを送信することに関連して説明した。次に、本明細書でSDTI-PFというもう1つの信号フォーマットについて述べる。このフォーマットSDTI-PFは、MPEG-2TSパケットを送信すると共に他の種類のパケットをSDTIを介してバッファリングなしで送信できるので、パケット・ジッタを減らすことができる。このようなパケットは、ATMセルや単方向インターネット・プロトコル(UDP)であってもよい。

【0052】この実施形態は、パケットストリームの送信について次のような幾つかの機能的特徴を有する。高いパッキング密度、データ・パケット・タイプ及び関連するパケット・メタデータの柔軟な配列、多数のデータ・パケット・タイプの多数チャンネルの送信、効果的なフォワード・エラー・コレクション(FEC)仕様の規定、精確なデータ・パケット出力タイミングを再生成する能力、組込みユーザ情報を加える能力、データ・

パケットに関する制御情報を、SDIスイッチングを検出するための連続カウンタの設置に使用できる、同じソース及び行先のアドレス間の多数のデータ・パケット・チャンネルの転送のためのチャンネル「ハンドル」を設けること、ストリーム内のデータ・パケットの位置を定めること。

【0053】SDIパラメータ

SMPTE305Mは、ビットレートが270Mbps及び360MbpsのSDIを指定しており、この実施形態は、その機器の仕様に従って270Mbpsのみか、又は両ビットレートで作動する。SDI-PPFは、図10に示すように、SDIの可変ブロック・モードを使用する。「セパレータ」及び「エンドコード」の間のスペースにおけるデータはすべて10ビットワードであり、そのデータは10ビットワードのうちビットb0～b7の中に入る8ビットを含み、ビットb8はビットb0～b7の偶数パリティにセットされ、ビットb9は奇数パリティにセットされる。

【0054】SDIヘッダの「ブロック・タイプ」ワードは、SMPTE305Mに従った可変ブロック・モードにセットされる。SDIデータ・タイプ・ワードは、データ・パケット・フォーマット（PF）の可変ブロック・ペイロードを示す値「11h」にセットされる。この実施形態は、SMPTE305Mのデータ拡張の便宜は使用しない。

【0055】SDIライン及びアドレス番号

各SDI可変ブロックにおけるデータは、ブロック・エンドまで必要な数のラインを通して続くので、SDIヘッダのライン番号は連続的でなければならない。また、SDIヘッダのソース及び行先のアドレス値は、任意の1つのSDI可変ブロックに関連する全部のラインの送信中は不変であることが必要である。

【0056】SDIスイッチング

SDIデータストリームの任意のスイッチングは、画像フレームの境界で生じるが、スイッチングの影響を軽減するための特殊な処理機器を使用することなくデータパケットの中身を良好にデコードする能力に影響する可能性がある。画像の切替えによって影響を受けるラインは、SMPTE-RP168に定められている。SMPTE-RP168に定められたラインは、過渡状態が起こるかも知れない切替えライン直前及び直後のラインと共に、使用しないことを勧める。連続カウンタは、切替えによって影響される可変ブロックを示すために設けてもよい。

【0057】TLDブロック構造

各可変ブロックのデータブロック領域は、図11に示すように、タイプ、長さ及びデータ（TLD）構造によって定められた1以上のブロックで満たされている。TLD構造の3つの構成部分は、次のとおりである。
タイプ：局所的1バイト・ラベルとしての「値」領域に

含まれるデータのタイプ、

長さ：上記値の長さ、

データ：タイプによって決まるデータ。

【0058】「タイプ」は、TLDブロックで送られるデータのタイプを識別するただ1つのバイトである。このタイプ値は、TLDパケットの一種（例えばMPEG-2TSパケット）か又はTLDメタデータの一種のいずれかを決定する。

【0059】TLDメタデータ・ブロックを受信すると、その値領域に含まれるメタデータは、そのSDI可変ブロックのエンドか又は同じタイプの新しいメタデータ構造のどちらか一方が受信されるまで、すべての後続TLDパケット・ブロックに適用される。TLDメタデータ・ブロックからのデータは、SDI可変ブロックを通して何の意味ももたないので、新しい各可変ブロックは、必要に応じてTLDメタデータの新しいセットを送信する。

【0060】図12は、本発明の実施形態によるSDI可変ブロックとTLDブロックの階層構造を示す。図12に示すように、各SDI「可変ブロック」のデータブロック領域は、TLDブロックですっかり満たされている。SDI「データブロック」の開始点、TLDブロック間、又は最後のTLDブロックとSDI「エンドコード」との間のいずれにも詰め物はない。

【0061】TLDのタイプ

TLDタイプは、TLDブロックにおけるデータのタイプを識別するために用いられる1バイトのワードである。タイプ値の割当てには、次のルールが適用される。「00h」のタイプ値は使用しない。「01h」から「0Fh」までの範囲内のタイプ値は、TLDメタデータ・ブロックを識別するのに用いられる。「10h」から「FFh」までの範囲内のタイプ値は、タイプワードの最大有効4ビットのみを用いてTLDパケットタイプを識別する。よって、これらのTLDタイプの15個のみを識別することができる。

【0062】「0Fh」より大きな値をもつタイプワードの最小有効4ビットは、次のとおりである。ビットb3は、TLDデータ領域のエンドにおけるFECの存在を識別するのに用いられる。b3＝「1」ならば、FECが存在し、そうでなければFECは存在せず、スペースは割当てられない。ビットb2は、データが組込みカウンタを含むかどうかを識別するのに用いられる。b2＝「1」ならば、組込みカウンタが存在し、そうでなければ組込みカウンタは存在せず、スペースは割当てられない。

【0063】ビットb1及びb0は、0～3の範囲内の2進値を構成する。該値＝「3」ならば、TLDデータ領域のヘッドに6ユーザ・バイトが含まれる。該値＝「2」ならば、TLDデータ領域のヘッドに4ユーザ・バイトが含まれる。

該値＝「1」ならば、TLDデータ領域のヘッドに2ユーザ・バイトが含まれる。

該値＝「0」ならば、TLDデータ領域のヘッドにユーザ・バイトは含まれない。

【0064】TLDの長さ

TLDブロックの長さは、TLDデータの長さを指定する。よって、この長さ値は、必要に応じ次のTLDブロックに飛ばすのに用いられる。TLD長さは、次のように定義される可変長ワードである。最初のバイトの値が「00h」～「FEh」の範囲内にあれば、その長さは最初のバイトの値によって与えられる。最初のバイトの値が「FFh」に等しければ、その長さ値は、すぐ後の2バイトの中に含まれる。したがって、254バイトより大きい長さのデータブロックをもつTLD構造は、最初のバイトが「FFh」の値に等しく、次の2つのバイトが「000h～FFFFh」の範囲の値をもつ長さを含む3バイトの長さになる。「FFFFh」の2バイトの長さ値は、将来可能性がある拡大のために取っておかれる。

【0065】TLDブロック

「OFh」までのタイプ値によって定められるTLDブロックは、以下のメタデータ定義の中で定められたフォーマットのメタデータを含む。「OFh」より上のタイプ値によって定められるTLDブロックは、以下のパケット定義の中で定められるようなデータパケットを含む。

【0066】データパケットは、次の構成部分より成る。

(1) 長さが0, 2, 4又は6バイトのユーザ・データ。該ユーザ・データの長さは、タイプワードのビットb0及びb1によって定められる。

(2) ジッタを無視できるデコーダ出力におけるデータパケットのタイミングを決めるのに使用できる3バイトの、小カウント値と大カウント値の組合せ。

(3) 6バイト長のリードソロモン・フォワード・エラー・コレクション(RS-FEC)。

【0067】このTLDパケットを図13に示す。最も簡単な場合、「0」にセットされた4つのLSBをもつTLDタイプ識別子があるデータパケットは、パケットデータのみを含み他の余計な成分を含まない簡単なデータパケットである。

【0068】ユーザ・データ領域

ユーザ・データのスペースは、0, 2, 4又は6バイトのユーザ・データ領域の余地がある2バイトのインクリメントで定義される。ユーザ・データ領域の中身は、個人的なデータであり、本明細書では定義しない。

【0069】パケット・タイミング・カウンタ

大小カウント値の組合せは、パケット・タイミング・カウンタを構成し、大小カウントワードによって夫々定められる整数部と分数部を有する。このパケット・タイミ

ング・カウンタは、データストリームの最初のパケットに対して正確なタイムでデコーダがパケットを出力できるように各パケットの開始点のタイミングを定めるものである。このパケット・タイミング・カウンタは、SDTIクロック周期と同程度に正確な出力パケットの開始位置のタイミングを与えることができる。

【0070】このパケット・タイミング・カウンタは、約30ミリ秒の周期を有する。パケット間の最大周期をこのパケット・タイミング・カウンタの周期より小さくして正確な動作を保証する。図5から図9までの説明は、正確な出力タイミングを制御するためパケット・タイミング・カウンタをどのように使用しうるかを説明するものである。

【0071】小カウント

小カウントは、前述した微細なスロット・サブカウントに対応するものである。8ビットの小カウントワードは、SDTIワード・クロックによって刻時させられるカウンタからサンプリングされた値である。小カウントワードのビットc0～c7は、小カウント値用に使われる。ビットc0は、小カウント値の最小有効ビットである。

【0072】270Mbps SDTIの場合、小カウント値は、27MHzクロック（これは、MPEG-2 TS PCRと同じレートである。）によって刻時させられ、0～11の範囲にわたってカウントし2.25MHzクロック周期を作るカウンタから格納される。360Mbps SDTIの場合、小カウント値は、36MHzクロック（これは、MPEG-2 TS PCRより高いレートである。）によって刻時させられ、0～15の範囲にわたってカウントし2.25MHzクロック周期を作るカウンタから格納される。小カウント・ワードに使用可能なビット数は、SDTIワード・クロックのレートを576MHzまで可能とするものである。

【0073】大カウント

大カウントは、前述した粗カウントに対応するものである。16ビットの大カウント・ワードは、2.25MHzクロックによって刻時させられるモード65536カウンタからサンプリングされた値である。大カウント値は、該カウント値のLSBからMSBまでを夫々表す、図13に示すようなビットc0～c15で構成される。

【0074】RS-FEC

RS-FECは、設けられている場合に、TLDタイプワードから最後のRS-FECワードまでの全TLD構造内容について補正を行う。RS-FECは、多数のTLDパケットの間にインタリーブされない。この実施形態では、エラー訂正はリードソロモンRS(L, L-6, T=3)として定義される。L<255の場合、RS-FECは元のRS(255, 249, T=3)コードから短縮されたコードである。

【0075】RSコードの生成多項式は、

$$RS(x) = (x \oplus \alpha^0) \cdot (x \oplus \alpha^1) \cdot (x \oplus \alpha^2) \cdot (x \oplus \alpha^3) \cdot (x \oplus \alpha^4) \cdot (x \oplus \alpha^5)$$

ただし、 α はガロア体GF(256)生成多項式より定義されるものである。

$$GF(x) = x^8 \oplus x^4 \oplus x^3 \oplus x^2 \oplus 1$$

【0076】エンコーダは3つのエラーを訂正する能力があるが、デコーダは、1つ又は2つのエラーのみを訂正して残りのエラー検出性能を保存することを選んでよい。また、RS-FECは、255バイト以下のTLD構造にのみ使用できるものであることに留意されたい。したがって、253より大きいTLD長さ値は、RS-FECの限界を越えるので、これには使用されない。

【0077】TLDブロックの定義

次の定義リストは、現在選ばれているものである。このリストを拡大してもよい。各TLD(タイプ、長さ、データ、またキー、長さ、値ともいう。)メタデータ定義としては、「タイプ」値によって定められる局部識別子と、SMPTEダイナミック・メタデータ・ディクショナリ内のメタデータ項目の場所を定める広域識別子との2つがある。両識別子は、同じメタデータ仕様を参照し易くするものである。タイプ値を短縮する理由は、SDTIトランスポートに使用される高速でデータを細かく調べ易くするためである。また、パッキング密度を高くしうるので、高速シリコンに対する記憶要件が簡単化される。しかし、この文書で特定した任意のメタデータ項目を拡大して、メタデータ・ディクショナリのもとになる十分なK-L-V構造を定義してもよいことに留意されたい。この十分に拡大されたK-L-V(key, length, value)構造は、異なる機器間のメタデータ項目の通常の交換のための基礎として使用できるであろう。

【0078】メタデータの定義

連続カウンタ

局部タイプ値＝「01h」

連続カウンタは、同じSDTIソース及び行先のアドレスを有するSDTI-PF可変ブロック毎に1つずつインクリメントする16ビットのモジュロ65536値である。このカウンタの目的は、信号路のスイッチングを検出するためである。ビット有効度は、LSBが先にある。即ち、LSBは最初のバイトのビットb0にあり、MSBは2番目のバイトのビットb7にある。

【0079】チャンネル・ハンドル

局部タイプ値＝「02h」

長さ＝2バイト

スロット・ハンドルは、同じSDTIソースと行先ヘッダのアドレスをもつが、異なるパケット・チャンネルを表すSDTI-PF可変ブロック内のパケットを識別するのに用いられる2バイトのワードである。スロット・

ハンドルは、同じSDTIソースと行先のアドレスに関連するすべてのパケットが単一チャンネルからのものである場合、ゼロにセットされる。2以上のチャンネルのパケット・ストリームがSDTI-PF可変ブロック上に多重化される場合、各チャンネルに対するスロット・ハンドル値は非ゼロである。この多重化されるチャンネルの数は、65535に制限される。SDTI-PF可変ブロックにチャンネル・ハンドル・メタデータが存在しない場合、そのTLDブロックは、1つのチャンネルからのものであると考えられる。

【0080】ストリーム位置インディケータ

局部タイプ値＝「03h」

長さ＝1バイト

パケット・ストリーム位置インディケータは、ストリーム内の次のTLDパケットの位置を識別するのに用いられる。ビットb2～b0は、パケット・ストリーム内の次のTLDパケットの位置を定める。これらの3ビットは、8つのストリーム状態を次のとおり識別する。

0＝ストリーム内のTLDパケット位置は未定である。

1＝TLDブロックは、ストリーム・スタート・パケットに先行するパケット(例えば、プリロール・パケット)であるストリーム・ヘッド・パケットを含む。

2＝TLDブロックは、ストリームの最初のパケットであるストリーム・スタート・パケットを含む。

3＝TLDブロックは、ストリーム・スタートとストリーム・エンドの2パケット間のパケットであるミッド・ストリーム・パケットを含む。

【0081】4＝TLDブロックは、ストリームの最終パケットであるストリーム・エンド・パケットを含む。

5＝TLDブロックは、ストリーム・エンド・パケットに続くパケット(例えば、ポストロール・パケット)であるストリーム・テール・パケットを含む。

6＝TLDブロックは、長さ1のストリームを示すストリーム・スタート・パケット及びストリーム・エンド・パケットの両方を含む。

7＝残してあるが未定義である。

ビットb7～b3は、残してあるが未定である。

【0082】TLDパケットの定義

次のパケット・タイプは、局部TLDタイプ・ワードとしてのみ定義されている。

MPEG-2トランスポート・ストリーム

タイプ値＝「8xh」

長さ＝組込みFECの存在及びタイプ並びにユーザ、カウンタ及びFECの構成部分の存在によって変化する。

MPEG-2トランスポート・ストリーム(MPEG-2 TS)パケットは、188バイトの長さである。フォワード・エラー・コレクション(FEC)が加えられると、パケット長はDVBエミッションの場合204バイトに、ATSCエミッションの場合208バイトに増加する。MPEG-2 TS FECは、パケットの間にインタリーブされ、エラー・バーストを生じるかも知れない伝送システムによる高度のデータロスとを予測したものである。極端な場合、SDTIは偶発的エラーを生じる可能性があるが、MPEG-2 TS FECは、そのインタリーブ長のためSDTIエラーの訂正には不適當であり、送信されるパケットの長さがたった188バイトであれば、このFECは存在しないことになる。したがって、SDTI-PFフォーマットは、SDTIリンクを通じて発生のある可能性があるエラーを訂正するため、任意のインタリーブされないFECを含んでいる。図2のa~cは、FEC能力が異なるMPEG-2 TSパケットを示す。SDTI-PF TSブロックは、必要に応じユーザデータ、タイミング再生、RS-FECの3つの機能をすべて使用してもよい。

【0083】SDTI-CP送信パケット

タイプ値=「9xh」

長さ=ユーザ、カウント及びFECの構成部分の存在によって変わる。

SDTI-CP送信パケットは、188バイトのMPEG-2トランスポート・ストリーム・パケットに基づくパケット構造を採用し、同一のやり方で管理される。

【0084】単方向インターネット・プロトコル・パケット(Uni-IP)

タイプ値=「Axh」

長さ=65535バイトまで可変である。

Uni-IPパケットは、RS-FECにより処理しうる長さを上回る長さをもつことができるので、Uni-IPパケットは通常FECを使用できない。しかし、Uni-IPパケットがソース装置により正しいFEC動作のために設定された限界より低く制約される場合、長さが254バイトより小さいパケットにのみ使用して成功するであろう。ユーザはまた、正しくパイプライン動作で動作するため幾つかのRSデコーダが固定パケット長に頼っていることに注意すべきである。よって、パケット長が変動するUni-IPパケットは、RS-FECが能動であれば、デコーダ問題を引き起こす可能性がある。

【0085】ATMセル

タイプ値=「Bxh」

長さ=53バイト

ATMセルは、53バイトの固定長を有するが、必要に応じこれにユーザ、カウント及びFECを加えてもよい。

【0086】以上本発明の実施形態を添付図面を参照して詳細に説明したが、本発明は、これらの実施形態に限定されるものではなく、特許請求の範囲に記載した本発明の範囲から逸脱することなく、当業者により種々の変更及び変形をこれらに加えるものである。

【0087】

【発明の効果】本発明によれば、パケット化した信号を送信するのに、従来のDVB-ASIを用いるよりはるかに柔軟性のある信号送信装置を提供することができる。

【図面の簡単な説明】

【図1】SDTIラインのペイロード領域の模式図である。

【図2】MPEG-2 TSパケットの模式図である。

【図3】スロットに分割され本発明の一実施形態によるフォーマットを有するSDTIペイロード領域を示す模式図である。

【図4】ペイロード・ヘッダ構造の模式図である。

【図5】本発明の一実施形態によるジッタの補正を説明する模式図である。

【図6】本発明によるSDTIシステムを介してMPEG-2 TSパケットを送信する装置の概略図である。

【図7】図6のタイミング・データ挿入回路及びタイミング補正回路の例を示す概略ブロック図である。

【図8】図7のカウンタ42の例を示す概略ブロック図である。

【図9】図7のカウンタ90の例を示す概略ブロック図である。

【図10】本発明の他の実施形態に用いられるSDTI可変ブロックの模式図である。

【図11】本発明の他の実施形態に用いられるTLDブロックの模式図である。

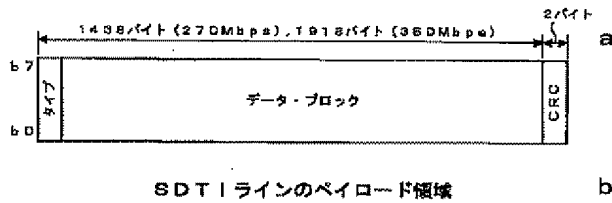
【図12】本発明の他の実施形態に用いられるSDTI可変ブロックとTLDブロックの階層構造を示す模式図である。

【図13】TLD複合パケットの模式図である。

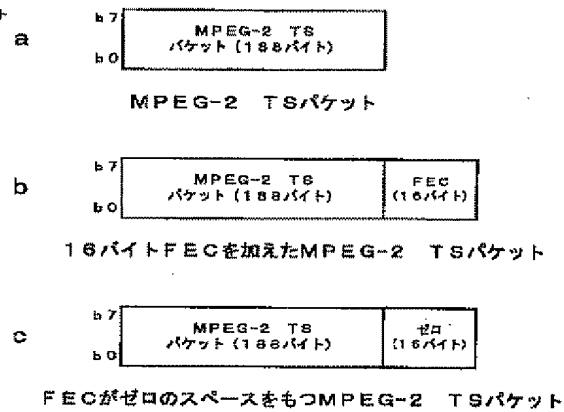
【符号の説明】

40……クロック、42……カウンタ、46……マルチプレクサ、80、92……モジュロnカウンタ、82、94……モジュロmカウンタ、84……デマルチプレクサ、88……比較手段、96……パケット出力手段、FEC……エラー訂正データ

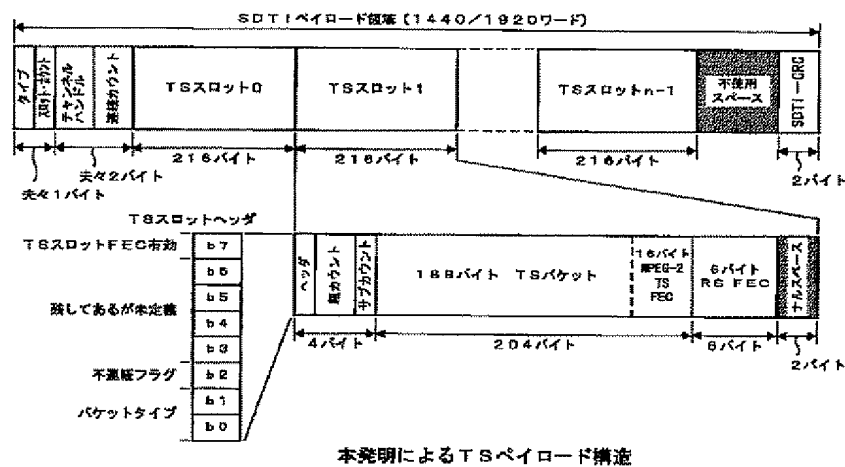
【図1】



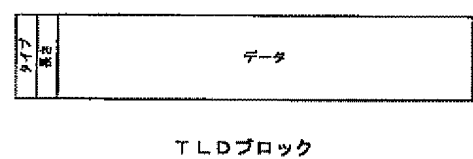
【図2】



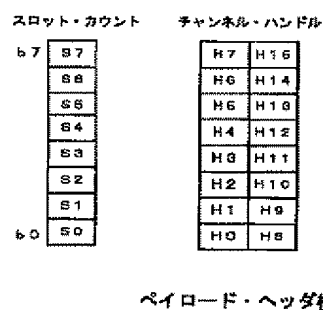
【図3】



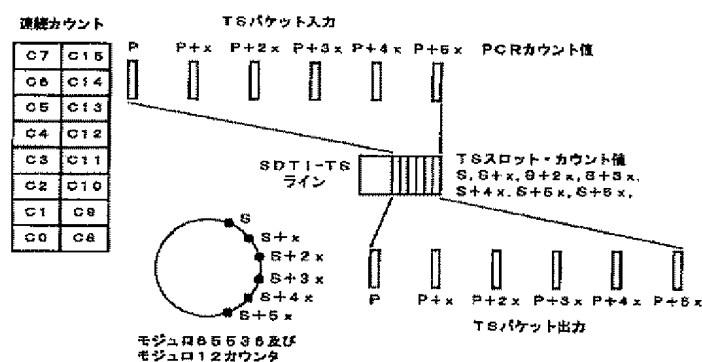
【図11】



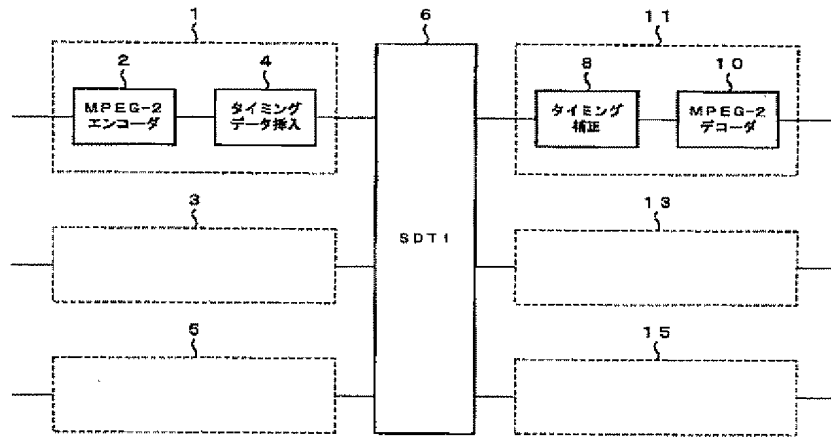
【図4】



【図5】

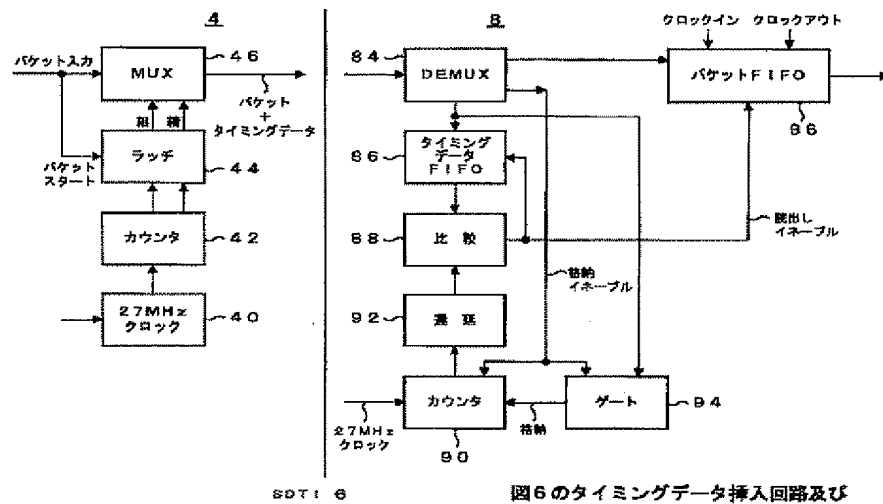


【図6】



本発明による信号送信装置

【図7】

図6のタイミングデータ挿入回路及び
タイミング補正回路の例

【図8】

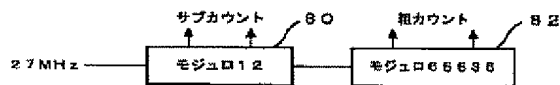


図7のカウンタ42の例

【図9】

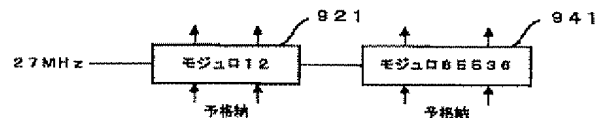


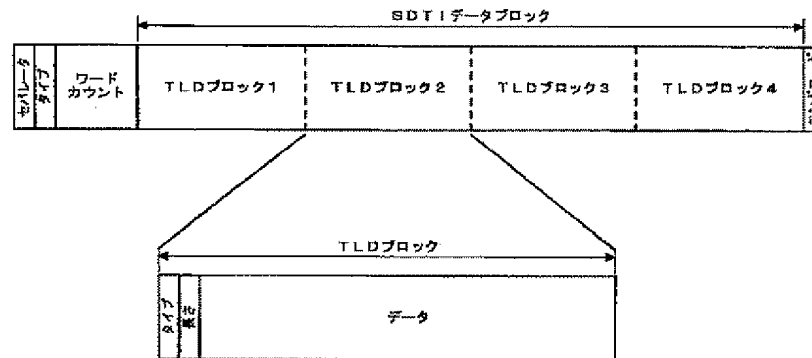
図7のカウンタ90の例

【図10】



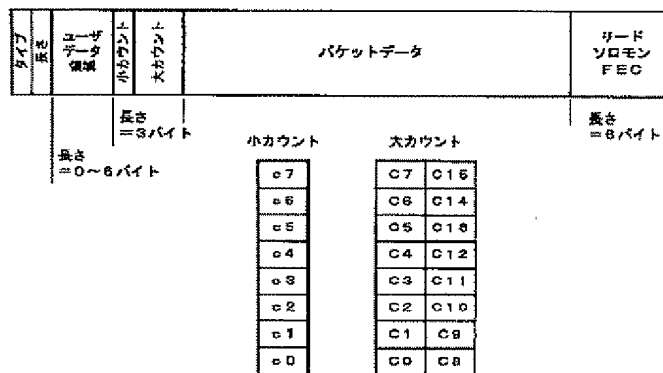
SDT1可変ブロックのフォーマット

【図12】



SDTI可変ブロックとTLDブロックの階層構造

【図13】



TLD複合パケットとその全構成部分

フロントページの続き

(72)発明者 オールディ、ステイーブン チャールズ
イギリス国 ハンプシャー、ページングス
トーク、ペンリス ロード 66

【外国語明細書】

1. Title of Invention

SIGNAL FORMAT, ENCODER, DECODER AND SIGNAL TRANSMISSION SYSTEM

2. Claims

1. An encoder for encoding a digital signal comprising data blocks, each data block including a header containing data relating to the block and a plurality of slots; each slot having a slot header relating to the slot and a data packet; the data packets containing successive parts of information from a source; a first slot which contains a first packet containing a first part of the said information from the source also containing a reference time; and the or each subsequent slot containing a subsequent packet of the information from the said source also containing timing information defining the timing of that packet relative to the reference time the encoder comprising a clock, and means for deriving from the clock a reference time defining the time of production of the said first packet and for providing the reference time information in the said first slot and for deriving from the clock the said timing information defining the times of production of the subsequent packages and providing the timing information in the subsequent slots as the subsequent packages are produced.

2. An encoder according to claim 1, wherein the timing information comprises coarse and fine timing information.

3. An encoder according to claim 2, wherein the clock comprises an input for receiving a clock signal, a modulo n counter which counts the clock signal and divides the clock signal frequency by n to produce the fine time information and a modulo m counter which counts the frequency divided clock signal produced by the modulo n counter, to produce the coarse time information, wherein m is much greater than n .

4. An encoding according to claim 3, wherein the clock signal frequency is $2.25n$ Mhz, where n is an integer.

5. An encoder according to claim 4, wherein the clock signal frequency is 27MHz, n is 12 and m is 65536.

6. An encoder according to claim 4, wherein the clock signal frequency is 36MHz, n is 16 and m is 65536.

7. An encoder according to claim 1, comprising a multiplexer for inserting the time information in the slots.

8. An encoder according to claim 1, further comprising means for inserting into the slot header a flag indicating whether the slot contains a said first packet.

9. An encoder according to claim 1, wherein the said dat blocks are variable length blocks.

10. An encoder according to claim 9, wherein the said slots are variable length slots.

11. An encoder according to claim 10, wherein the variable length slots comprise slots containing metadata and slots containing data described by the metadata.

12. An encoder according to claim 11, wherein a metadata slot precedes the data slots containing the data described by the metadata in the metadata slot.

13. An encoder according to claim 12, wherein the metadata slot contains metadata identifying a succeeding slot which contains a said first packet.

14. An encoder according to claim 11, wherein the variable length slots comprise a data field, a Type field containing data describing the type of data in the Data field and a Length Field defining the length of the data in the data field.

15. An SDTI system including an encoder according to claim 1.

16. A decoder for decoding a digital signal comprising data blocks, each data block including a header containing data relating to the block and a plurality of

slots; each slot having a slot header relating to the slot and a data packet; the data packets containing successive parts of information from a source; a first slot which contains a first packet containing a first part of the said information from the source also containing a reference time; and the or each subsequent slot containing a subsequent packet of the information from the said source also containing timing information defining the timing of that packet relative to the reference time the decoder comprising a clock, means for detecting the timing information of the packets, means for initially setting the clock to the said reference time on detection of the said first packet, means for comparing the clock time with the said timing information of the subsequent packets, and means for outputting the packets at the times when the timing information of the packets equals the clock time.

17. A decoder according to claim 16, wherein the timing information comprises coarse and fine timing information.

18. A decoder according to claim 17, wherein the clock comprises an input for receiving a clock signal, a modulo n counter which counts the clock signal and divides the clock signal frequency by n to produce the fine time information and a modulo m counter which counts the frequency divided clock signal produced by the modulo n counter, to produce the coarse time information, wherein m is much greater than n .

19. A decoder according to claim 18, wherein the clock frequency is 2.25MHz, wherein n is an integer.

20. A decoder according to claim 19, wherein the clock signal frequency is 27MHz, n is 12 and m is 65536.

21. A decoder according to claim 19, wherein the clock signal frequency is 36MHz, n is 16 and m is 65536.

22. A decoder according to claim 16, for use with a signal the slot header of which contains a flag indicating whether the slot contains a said first packet, the decoder comprising a demultiplexer for separating the flag and the packet, and means responsive to the flag for setting the clock to the reference time if the flag indicates a first packet.

23. A decoder according to claim 16, wherein the outputting means comprises a FIFO buffer.

24. A transmission system in which MPEG 2 TS packets are transmitted via an SDTI system.

25. A digital signal comprising: data blocks, each data block including a header containing data relating to the block and at least one slot ; each slot having a slot header relating to the slot and a data packet; the data packets containing successive parts of information from a source; a first slot which contains a first packet containing a first part of the said information from the source also containing a reference time; and the or each subsequent slot containing a subsequent packets of the information from the said source also containing timing information defining the timing of that packet relative to the reference time.

26. A signal according to claim 25, wherein the timing information comprises coarse timing information and fine timing information.

27. A signal according to claim 26, wherein the coarse and fine timing information are represented by separate words in the slot header.

28. A signal according to claim 25, wherein the slot header includes data indicating whether or not the packet is a first packet.

29. A signal according to claim 25, wherein the slot header includes data indicating packet type.

30. A signal according to claim 29, wherein the data indicating packet type indicates one or both of (i) packet length; and (ii) whether the packet includes active error correction data.

31. A signal according to claim 30, wherein each packet includes error correction data.

32. A signal according to claim 31, wherein the slot header includes data indicating whether or not the slot contains error correction data.

33. A signal according to claim 32, wherein each slot contains error correction data.

34. A signal according to claim 25, wherein the slots of each block are of fixed length and have predetermined positions in the block.

35. A signal according to claim 25, wherein the said data blocks are variable length blocks.

36. A signal according to claim 35, wherein the said slots are variable length slots.

37. A signal according to claim 36, wherein the variable length slots comprise slots containing metadata and slots containing data described by the metadata.

38. A signal according to claim 37, wherein a metadata slot precedes the data slots containing the data described by the metadata in the metadata slot.

39. A signal according to claim 38, wherein the metadata slot contains metadata identifying a succeeding slot which contains a said first packet.

40. A signal according to claim 36, wherein the variable length slots comprise a data field, a Type field containing data describing the type of data in the Data field and a Length Field defining the length of the data in the data field.

41. A signal according to claim 25, wherein the blocks and block headers conform to SDTI.

42. A signal according to claim 35, wherein the said packets are MPEG 2 TS packets or ATM cells or Internet Protocol Packets.

3. Detailed Description of Invention

BACKGROUND OF THE INVENTION

Field of the Invention

The present invention relates to signal processors and to a signal format. In particular the invention relates to a signal format, a signal encoder for encoding a signal according to the format, a corresponding decoder, and a signal transmission system including the encoder and decoder.

It is desired to transmit packetised signals such as MPEG 2 TS (Transport Stream) packets from one location or piece of equipment to another. It is known to transmit MPEG 2 TS packets via a DVB Asynchronous Serial Interface (ASI). DVB ASI is effective for the transmission of one transport stream from point to point such as between specific items of equipment but is otherwise relatively inflexible.

According to one aspect of the invention, there is provided a transmission system in which MPEG 2 TS packets are transmitted via an SDTI system.

Such a system provides greater flexibility than using DVB ASI. The SDTI (Serial Data Transport Interface) is defined in SMPTE 305M. SDTI transmits packets in a signal structure comprising frames of television lines. Ancillary data is carried in the horizontal blanking area of lines and data is carried in a payload area of each line. The payload area is in the active line interval. SDTI allows TS packets to be routed wherever SDI connections are available and also allows TS packets from more than one source to be transmitted. However, the carriage of TS packets over SDTI requires buffering to ensure that the packets are confined to the payload area of the SDTI and to allow multiple packets on each line for efficiency. The buffering process introduces delay and jitter (i.e. variation in the timing of the packets relative to each other) to the packets but, for accurate decoding of an MPEG 2 signal, the packets of that signal must be provided to the MPEG decoder with accurate timing relative to one another to allow correct decoding. Whilst absolute delay of packets is not a problem because it affects all packets equally, there is a need to correct jitter at or before the MPEG decoder.

Whilst the foregoing discussion describes the technical problem faced by the present invention with reference to the transmission of MPEG 2 TS packets via SDTI, similar problems may occur in the transmission of other types of time sensitive packets over other data transmission systems.

SUMMARY OF THE INVENTION

According to another aspect of the invention, there is provided a digital signal comprising: data blocks, each data block including a header containing data relating to the block and at least one slot; each slot having a slot header relating to the slot and a data packet; the data packets containing successive parts of information from a source; a first slot which contains a first packet containing a first part of the said information from the source also containing a reference time; and the or each subsequent slot containing a subsequent packet of the information from the said source also containing timing information defining the timing of that packet relative to the reference time.

According to another aspect of the invention, there is provided an encoder for encoding a digital signal comprising data blocks, each data block including a header containing data relating to the block and at least one slot; each slot having a slot header relating to the slot and a data packet; the data packets containing successive parts of information from a source; a first slot which contains a first packet containing a first part of the said information from the source also containing a reference time; and the or each subsequent slot containing a subsequent packet of the information from the said source also containing timing information defining the timing of that packet relative to the reference time, the encoder comprising a clock, and means for deriving from the clock a reference time defining the time of production of the said first packet and for providing the reference time information in the said first slot and for deriving from the clock the said timing information defining the times of production of the subsequent packages and providing the timing information in the subsequent slots as the subsequent packets are produced.

According to a further aspect of the invention, there is provided a decoder for decoding a digital signal comprising data blocks, each data block including a header containing data relating to the block and at least one slot; each slot having a slot header relating to the slot and a data packet; the data packets containing successive parts of

information from a source; a first slot which contains a first packet containing a first part of the said information from the source also containing a reference time; and the or each subsequent slot containing a subsequent packet of the information from the said source also containing timing information defining the timing of that packet relative to the reference time, the decoder comprising a clock, means for detecting the timing information of the packets, means for initially setting the clock to the said reference time on detection of the said first packet, means for comparing the clock time with the said timing information of the subsequent packets, and means for outputting the packets at the times when the timing information of the packets equals the clock time.

By providing the timing information in the signal, the decoder is enabled to output the packets with jitter substantially reduced and preferably eliminated to allow correct decoding in for example a subsequent MPEG decoder if the packets are MPEG TS packets. Packet jitter can corrupt the decoding. The decoder compares the timing information of each packet with an internal clock set by the reference time of the first packet and outputs the packets to, for example, a buffer when the clock time equals the packet time thus at least reducing the jitter.

In an embodiment of the invention, the blocks are payload areas in the active line intervals of SDTI signals which have ancillary data including address data in the non-active areas of the lines.

In an embodiment of the present invention as applied to SDTI, there is a fixed, integer, number of complete packets per SDTI line with padding (e.g. zeros) in any unused space. The slot lengths are thus fixed. The slots and packets on a particular SDTI line all have the same source and destination addresses and there is only one packet stream on that line.

In another embodiment of the present invention as applied to SDTI, the data blocks are variable length blocks which may occupy one or more lines. The slots within the variable length data blocks are variable length slots. Preferably, the slots are "TLD blocks" containing a Type field T containing data describing the type of data in the Data field D and a Length Field L defining the length of the data in the data field. Preferably the TLD blocks are of two types: metadata blocks; and data blocks. A metadata TLD block precedes the data TLD block(s) containing the data described by the metadata. The TLD data blocks contain the reference time and timing information. The metadata TLD block preceding those blocks contains data identifying which of the blocks contains the reference time.

For convenience, the Figures represent the data in parallel-format. In practice the data is transmitted in serial-format.

Overview

First Embodiments: SDTI-TS

The following description describes first examples of a novel signal format for carrying MPEG-2 Transport Stream packets over the SDTI (SMPTE 305M). MPEG 2 TS and SDTI are well known and will not be described herein in detail. The novel format is referred to herein as SDTI TS. SDTI uses television lines to carry data. The active line intervals are the pay load areas and contain the STD I data blocks. The non-active intervals of the lines contain ancillary data including the source and destination addresses of the data in the active line intervals. In accordance with an embodiment of the present invention, to ensure efficiency of transfer, the pay load area of each line of SDTI-TS is filled to capacity as much as possible with whole, fixed length, slots containing the packets from one Transport Stream. The novel format includes all information necessary to recover Transport Stream packets with a low jitter level at the SDTI-TS decoder and has the flexibility to meet a number of design requirements.

References

The following references, which are published standards, contain information relevant to the present invention.

- SMPTE 305M, Serial Data Transport Interface (SDTI).
- SMPTE RP 168, Definition of Vertical Interval Switching Point.
- ISO/IEC 13818-2: Information Technology - Generic Coding of Moving Pictures and Associated Audio Information: Video, (MPEG-2).
- ISO/IEC 13818-2: Information Technology - Generic Coding of Moving Pictures and Associated Audio Information: Systems, (MPEG-2).

DVB: Interfaces for CATV/SMATV Headends and Similar Professional Equipment;

Asynchronous Serial Interface (ASI).

SMPTE 259M, 10-bit 4:2:2 Component and 4fsc NTSC Composite Digital Signals/Serial Interface.

SMPTE 291 M, Ancillary Data Packet and Space Formatting.

Background

The currently established interface for the carriage of MPEG-2 Transport Stream (TS) packets is the DVB ASI which places packets within a small tolerance of the position required to ensure minimal impact on the decoder buffer. However, ASI cannot easily carry more than one TS, neither can ASI be supported by commonly available SDI based equipment. It is useful as a specialised point-to-point interface between specific items of equipment.

Illustrative examples of the present invention

One aspect of the present invention proposes the carriage of MPEG2 TS packets over the SDTI. This allows a more general approach to connectivity by allowing TS packets to be routed wherever SDI connections are available. However, the carriage of TS packets over the SDTI requires buffering to ensure that packets are confined to the payload area of the SDTI and to allow multiple TS packets on each line for efficiency. The result of this buffering process is to introduce both a delay and jitter to the packet stream. The delay is not a problem which concerns the present invention. The examples of the present invention described herein allow the packet jitter to be reduced to insignificant levels. As a benchmark, the present embodiments allow a DVB-ASI input to be carried through SDTI-TS and decoded to create a new DVB-ASI signal with minimum additional jitter. The SDTI-TS can also operate directly from other interface points as required.

SDTI PARAMETERS

SMPTE 305M specifies the SDTI for bit rates of 270Mbps and 360Mbps and examples of the invention may operate on both bit rates and at higher bit rates. The examples of SDTI-TS described herein use the fixed block size mode of the SDTI with one block per line. The format of each fixed length block is shown in Figure 1. The

block is either 1438 or 1918 bytes long including a ten bit SDTI-TYPE word as described hereinafter. The block typically includes an additional 2 bytes of error detection CRC for a total of 1440 or 1920 bytes as shown in Figure 1.

TYPE WORD

The SDTI data block has a block type word.

For 270Mbps SDTI, the Block Type value is "01h" which specifies a block size of 1438 words per line (including the Type value).

For 360Mbps SDTI, the Block Type value is "09h" which specifies a block size of 1918 words per line (including the Type value).

The SDTI Data Type word has a suitable value: there is no assigned value at present.

The input format is 8 bit data entered into bits b7 -b0 of a 10 bit word. Bit b8 of the 10 bit word is the even parity of bits b7-b0 and bit b9 is the complement of bit b8.

The optional FEC (Forward Error Correction) may be added to the block to give added data security. Whether FEC is added or not, bits b7 and b6 of the Block Type word are set according to SMPTE 305M.

The present examples of the invention do not use the Data Extension facility of SMPTE 305M. It is recommended to avoid use of the SDI switching lines as defined in SMPTE- RP 168

MPEG-2 TRANSPORT STREAMS

As shown in Figure 2a, MPEG-2 TS (Transport Stream) packets are 188 bytes in length. As shown in Figure 2b if DVB Forward Error Correction (FEC) is added, then the packet length is 204 bytes. The MPEG-2 TS FEC is interleaved across packets and designed for correcting serious data loss through transmission errors which may be bursty. Whilst in extreme cases SDTI can create occasional errors, the MPEG-TS FEC is unsuitable for correcting SDTI errors due to the interleave length and this FEC will not exist if the packets carried are only 188 bytes in length. Consequently,

the SDTI-TS format of an embodiment of the invention includes an optional FEC to correct for any errors which may occur through the SDTI link.

PAYLOAD FORMAT

In accordance with the first embodiments of the invention, the MPEG-2 TS packets are wholly contained in an SDTI fixed block placed in a TS Payload format an example of which is described as follows with reference to Figure 3. The illustrative TS Payload format has a 2 layer structure as indicated in Figure 3. The payload has a TS Payload header defining parameters which apply to the whole payload line, and a fixed number of TS Slots into which the MPEG2 TS packets are placed. Each TS Slot has a slot header which defines parameters associated with the TS packet in the slot, followed by the TS packet itself and terminated by an optional SDTI RS-FEC of 6 bytes.

TS Payload Structure

The SDTI Type word of the TS Payload is the type word at the left hand side of figure 1. The bit assignments of the Slot Count, Channel Handle and Continuity Count words are shown in Figure 4.

TS Payload Slot Count

The Slot Count is an 8 bit word which defines the number of TS Slots for this Payload line. The number of TS Slots has a lower value of 1 and an upper value defined by the payload length. In the case of 270Mbps SDTI, the maximum number of TS Slots is 6. In the case of 360Mbps SDTI, the maximum number of TS Slots is 8. Higher SDTI rates can support appropriately higher numbers of TS Slots in the payload area. In the present embodiments of the invention, data space between the last TS Payload Slot and the SDTI CRC words is not used for any other purpose or application. Lower values of payload slot count tend to reduce the codec delay. Further information on this topic is given herein below.

TS Payload Channel Handle

The Channel Handle is a 16 bit word organised as two 8 bit words with the most significant word first. This word is used to distinguish between TS Payload lines having the same SDTI source and destination header addresses but representing different channels of TS transmission. In the present embodiments the Channel Handle value is set to "0000h" but other values may be chosen.

TS Payload Continuity Count

The "Continuity Count" is a modulo 65536 count which increments by 1 for every TS Payload line having the same SDTI Source and Destination addresses and the same Channel Handle value. The purpose of this count is to provide detection of signal path switching.

TS Slot Structure

In the present embodiments every TS Slot is 216 words in length and TS Slots are arranged in continuous order immediately following the TS payload header.

A TS Slot contains information in the following sequence:

- a TS Slot header of 2 word length,
- a coarse count value of 2 words in length clocked at 2.25MHz,
- a fine count which is one word in length clocked at a multiple of 2.25MHz.
- a 188 byte TS Packet,
- a 16 byte space for the optional MPEG-2 TS FEC,
- a 6 byte space for the optional SDTI-TS FEC and
- a 2 byte null data space.

TS Slot Header

The TS Slot header is an 8 bit word with bits b7-b0 defined as follows:

Bit b7 defines whether the TS Slot has an active SDTI FEC.

b7 is "0" if the SDTI FEC is not active and is " 1 " if the SDTI FEC is active.

Bits b6-b3 are reserved but not defined.

Bit b2 is a TS discontinuity flag. If the TS packet in this TS slot is not related to the previous TS packet, the b2 is set to "1", else it is set to "0". A discontinuous TS packet (i.e. b2=1) also indicates that this is the first packet in a new TS packet sequence.

Bits b1 and b0 form a code which define the TS packet type as follows:

b1	b2	TS Type
0	0	188 byte TS packet with no 16 byte FEC.
0	1	Reserved but not defined.
1	0	204 byte TS packet with an inactive 16 byte FEC
1	1	204 byte TS packet with an active 16 byte FEC.

TS Slot Coarse Count

The coarse count value is derived from a modulo 65536 counter clocked at 2.25MHz at the SDTI-TS encoding point and defines the coarse timing of the TS packet for the decoder to be able to output the TS packet at the correct time relative to the previous TS packet. This coarse timing is used in conjunction with the next word, the TS Slot sub count which contains a fine timing derived from a clock at an integer multiple of 2.25MHz to provide a clock with a reference timing close to that of the TS Program Clock Reference (PCR). The coarse and fine timing are preferably derived from the same master clock which is for example the PRC clock operating at 27 MHz.

It is not required for the 2.25MHz counter to match the period of the PCR of 27 hours. The 2.25MHz counter only needs a sufficient cycle period guaranteed to be longer than the maximum period between TS packets. The 2.25MHz cycle period is approximately 30msecs. The 2.25MHz counter increments by 1 for every 12 increments of the PCR.

TS Slot Sub Count

In the case of 270Mbps SDTI, the sub count is clocked by a 27MHz clock (which is at the same rate as the PCR) and counts over the range 0 to 11. In the case of 360Mbps SDTI, the sub count is clocked by a 36MHz clock (which is at a higher rate than the PCR) and counts over the range 0 to 15. The number of bits available for the Sub Count value allows SDI word (parallel) clock rates up to 576MHz. How the TS Slot counts can be used for output timing control is described herein later.

TS Packet Data

The 204 byte space for the TS packet is a fixed allocation. The contents of this space are filled with 188 byte or 204 byte TS packet data as defined by bits b1 and b0 of the TS slot header word. In the cases of 188 byte TS packets and of 204 byte TS packets with inactive TS FEC, the 16 byte TS FEC space is null filled.

TS Slot RS FEC

The FEC is applied over the 214 bytes of the TS slot from the 2.25MHz count word to the last word of the FEC. If the TS slot header word FEC valid bit (b7) is set to "0", then all 6 words of the FEC are set to "00h". The FEC is a Reed-Solomon, R-S(214, 208, T=3) shortened code from the original R-S(255, 249, T=3) code.

The R-S code generator polynomial is:

$$R - S(x) = (x \oplus \alpha^0).(x \oplus \alpha^1).(x \oplus \alpha^2).(x \oplus \alpha^3).(x \oplus \alpha^4).(x \oplus \alpha^5)$$

where α is defined by Galois Field GF(256) generator polynomial:

$$GF(x) = x^8 \oplus x^4 \oplus x^3 \oplus x^2 \oplus 1.$$

TS Slot Null Space

The TS slot ends with 2 words of zero filled null space.

USING THE TS SLOT TIMING DATA

Figure 5 illustrates the SDTI-TS coding process from the MPEG-2 TS packet stream input to output through the SDTI-TS format. The combination of the TS Slot 2.25MHz count and the sub count values defines a count value with a resolution at least as high as the PCR and with a period before repetition of approximately 30msecs. On reception of the first TS packet in a sequence, the decoder may output the packet whenever it is ready and, at the same time, sets an internal counter running at the SDTI word rate (27/36MHz) to the TS slot count value. This counter counts in the same

manner as defined for the 2.25MHz count and sub count. Thereafter, for all remaining packets, the decoder outputs a packet when the decoder counter equals the counter stored in the TS slot count thus guaranteeing the integrity of the output TS packet timing. In the case of 270Mbps SDTI, the count value clocks at the same rate as the TS packet PCR and should result in no timing jitter. In the case of 360Mbps SDTI, the count increments by 4 for every 3 increments of the TS packet PCR, therefore, resulting in plus or minus 1 PCR clock jitter. This value is well within the MPEG-2 defined limits.

SYSTEM DELAY CONSIDERATIONS

The payload area of each SDTI line is packed with as many complete slots and thus TS packets as possible. In the present embodiments the slots are of fixed length and the SDTI payload area contains only complete slots. Spare space in a payload area is filled with zeros. This ensures that the waste of SDTI lines is minimised and thus the maximum number of lines are available for the transfer of other data. The following examples are based on 270Mbps SDTI-TS. At a bit rate of 4Mbps, approximately 15 lines per frame (525/30) are used for the carriage of MPEG-2 TS packets and the codec delay is just over 2msec. At a bit rate of 50Mbps, approximately 185 lines per frame (525/30) are used for the carriage of MPEG-2 TS packets and the codec delay is thus reduced to around 200usec. To a first approximation, the codec delay is inversely proportional to the bit rate. Thus at low bit rates, delay can only be reduced by reducing the number of TS packets per line and thus occupying proportionately more SDTI lines. If the 4Mbps example above occupied only 1 TS packet per line, then the delay would be reduced to approximately 400 usec.

Example of an SDTI TS system

Referring to Figure 6 there is shown an example of an SDTI TS system. The system comprises a data source 1. The data source 1 includes an MPEG2 encoder 2 which is known and which produces known MPEG 2 TS packets. The packets may or may not include the FEC data, as shown in Figures 2a to 2c. The packets are outputted from the MPEG encoder with the correct relative timing: i.e. they are relatively jitter-

free. A timing data inserter 4 of the source 1 (and described in more detail with reference to Figures 7 and 8) inserts the coarse and fine timing data into a header. The timing data defines the correct timing of the packets relative to a reference time. The MPEG2 TS packets with the timing data are fed to the SDTI 6. The SDTI 6 also receives packets and other data from other sources 3 and 5. The other sources may or may not be the same as source 1. The SDTI routes the data from the sources 1, 3 and 5 in known manner to respective corresponding destinations 11, 13 and 15. For that purpose one SDTI line has a source address and a destination address in its ancillary data and carries only data from the addressed source to the addressed destination. As described above the MPEG2 TS packets are buffered in the SDTI so that they incur jitter. At the destination 11 the packets from source 1 are subject to timing correction in corrector 8 before delivery to an MPEG decoder with the correct timing. An example of the corrector 8 is described with reference to Figure 7.

The SDTI assembles the MPEG2 TS packets into the slots shown in Figure 3 adding the slot header data to the timing data and adding the block header data, as also shown in Figure 3, to the block. The SDTI also adds the 6 byte FEC, the zeros for the null space and the optional SDTI CRC. The SDTI also determines when a packet is the first packet of a new packet sequence and sets the bit b2 of the TS discontinuity flag to '1'.

Referring to Figure 7, an example of the timing data inserter comprises a 27MHz clock 40 which clocks a counter 42 to produce the timing data. The clock 40 is derived from the SDTI clock. An example of the counter is shown in Figure 8. The counter comprises a Modulo 12 counter 80 clocked at 27MHz which produces the fine timing data and which produces a 2.25MHz clock for clocking a Modulo 65536 counter 82 which produces the coarse timing data. The counters 80 and 82 are free running. A latch 44 temporarily stores the fine and coarse counts. Upon receiving a packet start signal it feeds the current counts to a multiplexer 46 which inserts the timing data into the bit stream at the start of the packet. The first packet of a packet sequence is allotted whatever arbitrary time is indicated by the counter 42 at the moment of its production. That time is a reference time for the subsequent packets of

the sequence. The coarse counter 82 sequences through its counts with period of $65536/(2.25 \times 10^6)$ seconds or about 30 ms.

At the destination, an example of the corrector 8 comprises a demultiplexer 84 which detects the slot header and determines therefrom whether a packet is the first in a sequence as indicated by the discontinuity flag ($b2=1$) and the fine and coarse timing data. The demultiplexer separates the MPEG2 TS packet from the slot header and feeds the packet to a FIFO buffer 96. The FIFO buffer 96 stores one SDTI block of packets. The packets are on average clocked into and out of the FIFO at the same rate e.g. 270 M bits per second as they are originally produced at the source 1. However, the packets may be input irregularly but are output continuously with correct timing. The demultiplexer feeds the timing data to a comparator 88 via another FIFO 86 and directly to a gate 94. If the packet is a first packet, the discontinuity flag $b2=1$ enables the gate 94 to load the timing data, which represents the reference time, into the counter 90, thereby setting the counter to the reference time.

The counter 90 an example of which is shown in Figure 9 comprises a modulo 12 counter 921 and a modulo 65536 counter 941 arranged identically to the corresponding counter 80 and 82 of Figure 8 and counting a 27MHz clock provided by the SDTI. The clock provided by the SDTI is synchronous in known manner with the source clock 40. The counters 921 and 941 differ from the counters 80 and 82 in that they can be preloaded with the reference count. Once loaded the counters 921 and 941 are free-running in the same way as the counters 80 and 82 and are thus in delayed synchronism with them. The comparator 88 compares the timing data from the demultiplexer with the timing data from the counter 90. When the compared timing data are equal, the comparator enables the FIFO 96 to output a packet with the correct timing.

The packets and the timing data are separated by the demultiplexer 84 and fed into the FIFOs 86 and 96. The FIFOs contain corresponding sequences of packets and timing data. As the packets are moved through the FIFO 96 to its output, so are the corresponding timing data moved through the FIFO 86 to maintain correspondence with the packets in the FIFO 96. After each item of timing data reaches the output of the FIFO 86 that item is cleared from the FIFO. Each item of timing data at the output

of the FIFO 86 is compared with the count in the counter 90 and when the count represented by the timing data equals the count in the counter, a read out enable signal causes the FIFO 96 to start reading out the corresponding packet which accordingly has the correct timing. The enable signal also clears the timing data from the output of the FIFO 86 and moves the next item of timing data to the output to be compared with the count in the counter 90.

Second Embodiments: SDTI-PF

Figures 1 to 4 describes a signal format referred to herein as STDI-TS which uses the fixed block size mode of the SDTI will one block per line. That signal format is described in the context of transporting MPEG 2 bit streams using the SDTI. The following description describes another version of the signal format referred to herein as SDTI-PF. This format SDTI-PF allows MPEG2-TS packets to be transported and also allows for other kinds of packet to be carried over the SDTI, with or without buffering to reduce packet jitter. Such packets may be ATM cells and packets based on the Unidirectional Internet Protocol (Uni-IP).

This embodiment provides several features for the carriage of packet streams such as:

- A high packing density.
- A flexible arrangement of data packet types and associated packet metadata.
- Carriage of multiple channels of multiple data packet types.
- Provision for a powerful Forward Error Correction (FEC) specification.
- The ability to reproduce accurate data packet output timing.
- The ability to add embedded user information.
- The control information associated with data packets can be used to:
 - Provide a continuity counter to detect SDI switching.
 - Provide a channel 'handle' for transfers of multiple channels of data packets between the same source and destination addresses.
 - Define the position of a data packet in a stream.

SDTI PARAMETERS

SMPTE 305M specifies the SDTI for bit rates of 270Mbps and 360Mbps and this embodiment may operate on either 270Mbps only, or on both bit rates according to the equipment specification. The SDTI-PF uses the Variable Block mode of SDTI as shown in figure 10. The 'Separator' and 'End Code' words are special 10 bit values defined in SMPTE 305M. All data in the space between the 'Separator' and 'End Codes' are 10 bit words where

- the data comprises 8 bits entered into bits b0 to b7 of the 10 bit word,
- bit b8 is set to even parity of bits b0 to b7, and
- bit b9 is set to be odd parity.

The 'Block Type' word of the SDTI header is set to Variable Block mode according to SMPTE 305M.

The SDTI Data Type word is set to the value '11h' indicating the Data packet Format (PF) variable block payload.

This embodiment does not use the Data Extension facility of SMPTE 305M.

SDTI LINE AND ADDRESS NUMBERS

Since the data in each SDTI variable block may continue through as many lines as necessary until the block end, it is necessary that the SDTI header line numbers are contiguous. It is also necessary that the SDTI header source and destination address values are constant throughout the transmission of all lines associated with any one SDTI variable block.

SDTI SWITCHING

The arbitrary switching of SDTI data streams, although at the picture frame boundary, may affect the ability to successfully decode the contents of data packets without the use of special processing equipment to mitigate the switching effects. The lines affected by a picture switch are defined in SMPTE RP168. It is recommended to avoid the use of the lines defined by SMPTE RP168 together with the lines immediately prior to and following the switch line where transient conditions may occur. A continuity count can be provided to indicate variable blocks affected by a switch.

TLD BLOCK STRUCTURE

The Data Block area of each variable block is filled with one or more blocks defined by a Type, Length and Data (TLD) construct as shown in figure 11.

The three components of the TLD construct are:

Type: the type of data contained in the 'value' area as a local 1 -byte label,

Length: the length of the value, and

Data: the data as defined by the Type.

'Type' is a single byte which identifies the type of data carried in the TLD block. The Type value may identify either one kind of TLD packet (such as an MPEG-2 TS packet) or one kind of TLD metadata.

When a TLD metadata block is received, the metadata contained in the value area applies to all subsequent TLD packet blocks until either the end of the SDTI variable block or until a new TLD metadata construct of the same type is received. The data from a TLD metadata block does not carry any significance cross SDTI variable blocks, so each new variable block carries a new set of TLD metadata as needed.

Figure 12 illustrates a hierarchy of SDTI variable blocks and TLD blocks according to an embodiment of the present invention. As shown in figure 12 the data block area of each SDTI 'Variable Block' is wholly filled with TLD blocks. There is no padding either at the start of the SDTI 'Data Block', or between TLD blocks or between the end of the last TLD block and the SDTI 'End Code'.

TLD TYPE

The TLD Type is a 1 byte word used to identify the type of data in the TLD block.

The following rules apply to the assignment of Type values:

A Type value of '00h' is not used.

Type values in the range '01h' to '0Fh' are used to identify TLD metadata blocks.

Type values in the range from '10h' to 'FFh' use only the most significant 4 bits of the Type word to identify TLD packet types. Thus only 15 of these TLD types can be identified.

The least significant 4 bits of Type words with a value greater than '0Fh' are as follows:

Bit b3 is used to identify the presence of an FEC at the end of the TLD data area. If b3 = '1' then the FEC is present, else the FEC is not present and no space is allocated.

Bit b2 is used to identify if the data contains an embedded counter. If b2 = '1', then the embedded counter is present, else the embedded counter is not present and no space is allocated.

Bits b1 and b0 form a binary value in the range 0 to 3.

If the value = '3', then 6 user bytes are contained at the head of the TLD data area.

If the value = '2', then 4 user bytes are contained at the head of the TLD data area.

If the value = '1', then 2 user bytes are contained at the head of the TLD data area.

If the value = '0', then no user bytes are contained at the head of the TLD data area.

TLD LENGTH

The Length of a TLD block specifies the length of the TLD Data. Thus the Length value can be used to skip to the next TLD block if needed.

The TLD Length is a variable length word defined as follows:

If the value of the first byte lies in the range '00h' to 'FEh', then the length is given by the value of the first byte.

If the value of the first byte is equal to 'FFh' then the Length value is contained in the 2 bytes which immediately follow.

Thus TLD constructs with data blocks having a length of greater than 254 bytes result in a length of 3 bytes of which the first byte is equal to the value of 'FFh' and the

next two bytes contain a Length whose value may range from '0000h' to 'FFFEh'. The 2-byte Length value of 'FFFFh' is reserved for future possible expansion.

TLD BLOCK

TLD blocks defined by a Type value up to '0Fh' contain metadata of a format defined in the Metadata Definitions below.

TLD blocks defined by a Type value above '0Fh' contain data packets as defined in the Packet Definitions below.

A Data packet may be formed from the following components:

User data of length 0, 2, 4 or 6 bytes. The length of the user data is defined by bits b0 and b1 of the Type word;

A Minor Count value in combination with a Major Count value giving 3 bytes which can be used to re-time the data packets at the decoder output with negligible jitter; and

A Reed-Solomon Forward Error Correction (RS-FEC) of 6 bytes length.

The TLD packet is shown in figure 13. At its simplest level, a data packet with a TLD Type identifier which has the 4 LSBs set to '0' becomes a simple data packet containing only the packet data and no extra components.

USER DATA AREA

User data space is defined in increments of 2 bytes allowing 0, 2, 4 or 6 bytes of User Data area. The contents of the User Data area are private data and not defined herein.

PACKET TIMING COUNTER

The combination of Major and Minor Count values forms the Packet Timing counter with integer and fractional parts defined by the Major and Minor Count words respectively. The Packet Timing counter defines the timing of the start of each packet for a decoder to be able to output the packet at the correct time relative to the first packet in any stream. The Packet Timing counter can provide output packet start positions timings which are as accurate as the SDTI clock period.

The packet timing counter has a cycle period of approximately 30msecs. The maximum period between packets is less than the packet timing counter cycle period to guarantee correct operation.

The description of figures 5 to 9 describes how the Packet Timing counter can be used for accurate output timing control.

MINOR COUNT

The minor count corresponds to the fine slot sub count described hereinabove.

The 8-bit Minor Count word is a value sampled from a counter clocked by the SDTI word clock. Bits c0 to c7 of the Minor Count word are used for the minor count value. Bit c0 is the least significant bit of the minor count value.

In the case of 270Mbps SDTI, the minor count value is loaded from a counter clocked by a 27MHz clock (which is at the same rate as the MPEG-2 TS PCR) and counting over the range 0 to 11 to create a 2.25MHz clock period.

In the case of 360Mbps SDTI, the minor count value is loaded from a counter clocked by a 36MHz clock (which is at a higher rate than the MPEG-2 TS PCR) and counts over the range 0 to 15 to create a 2.25MHz clock period.

The number of bits available in the Minor Count word allows SDTI word clock rates up to 576Mhz.

MAJOR COUNT

The major count corresponds to the coarse count described hereinabove. The 16-bit Major Count word is a value sampled from a modulo 65536 counter clocked by the 2.25MHz clock. The Major Count value is formed from bits C0 to C15 as shown in figure 4 representing, respectively, the LSB to the MSB of the count value.

R-S FEC

Where provided, the R-S FEC provides correction for the whole TLD structure content from the TLD Type word to the last R-S FEC word. The R-S FEC is not interleaved over multiple TLD packets. In this embodiment, the error correction is

defined as a Reed-Solomon R-S (L, L-6, T=3). Where L<255, then the RS-FEC is a shortened code from the original R-S (255, 249, T=3) code.

The R-S code generator polynomial is:

$$R - S(x) = (x \oplus \alpha^0).(x \oplus \alpha^1).(x \oplus \alpha^2).(x \oplus \alpha^3).(x \oplus \alpha^4).(x \oplus \alpha^5)$$

where α is defined by Galois Field GF(256) generator polynomial:

$$GF(x) = x^8 \oplus x^4 \oplus x^3 \oplus x^2 \oplus 1.$$

Note that although the encoder provides the capability of correcting 3 errors, a decoder may choose to correct only 1 or 2 errors to preserve residual error detection performance.

Note also that the RS-FEC can only be applied to TLD structures of 255 bytes or less. Thus a TLD length value of greater than 253 will exceed the RS-FEC limit and thus is not applied.

TLD BLOCK DEFINITIONS

The following list of definitions is that currently preferred. Extensions to this list may be made.

Each TLD, (Type, Length, Data, also referred to as Key, Length, Value) metadata definition has both a local identifier defined by the 'Type' value and a global identifier which defines the place of the metadata item in the SMPTE Dynamic Metadata Dictionary. Both identifiers are referencing *the same* metadata specification. The reason for the shortened Type value is for ease of parsing the data at the high speeds used by the SDTI transport. There is also a gain in packing density and hence simplified storage requirements on high speed silicon. But it should be noted that any metadata item specified in this document may be expanded to define the full K-L-V construct on which the Metadata Dictionary is based. This fully expanded K-L-V (key,

length, value) construct may then be used as a basis for the common interchange of metadata items between different applications.

METADATA DEFINITIONS

CONTINUITY COUNT

Local Type value '01h'

The Continuity Count is a 16 bit, modulo 65536 value which increments by 1 for every SDTI-PF variable block having the same SDTI Source and Destination addresses. The purpose of this count is to provide detection of signal path switching.

The bit significance is LSB first. Thus, the LSB lies at bit b0 of the first byte and the MSB lies at bit b7 of the second byte.

CHANNEL HANDLE

Local Type value = '02h'

Length = 2 bytes.

The Slot Handle is a 2 byte word which is used to distinguish between packets within a SDTI-PF variable block having the same SDTI source and destination header addresses but representing different packet channels. A Slot Handle is set to zero where all packets associated with the same SDTI Source and Destination Addresses are from a single channel. Where 2 or more channels of packet streams are multiplexed onto SDTI-PF variable blocks, the Slot Handle values for each channel is non-zero. The number of multiplexed channels is limited to 65535.

If the Channel Handle metadata is not present in SDTI-PF variable block, then the TLD blocks are considered to be from one channel.

STREAM POSITION INDICATOR

Local Type value '03h'

Length = 1 byte.

The packet stream position indicator is used to identify the position of the following TLD packet in a stream.

Bits b2 to b0 define the position of the following TLD packet in a stream of packets. These 3 bits identify 8 stream states as follows:

0 = the TLD packet position in a stream is undefined.

1 = the TLD block contains a stream head packet which is any packet which precedes the stream start packet (e.g. pre-roll packets)

2 = the TLD block contains a stream start packet which is the first packet of a stream.

3 = the TLD block contains a mid-stream packet which is any packet between the stream start and stream end packets.

4 = the TLD block contains a stream end packet which is the last packet of a stream.

5 = the TLD block contains a stream tail packet which is any packet which follows the stream end packet (e.g. post-roll packets).

6 = the TLD block contains both a stream start packet and a stream end packet signifying a stream of length 1.

7 = reserved but undefined.

Bits b7 to b3 are reserved but undefined.

TLD PACKET DEFINITIONS

The following packet types are defined only by the local TLD Type Word.

MPEG-2 TRANSPORT STREAMS

Type value = '8xh'

Length = variable depending on the presence and type of embedded FEC together with the presence of User, Count and FEC components.

MPEG-2 Transport Stream (MPEG-2 TS) packets are 188 bytes in length. If Forward Error Correction (FEC) has been added, then the packet length is increased to 204 bytes for DVB emission and 208 bytes for ATSC emission. The MPEG-2 TS FEC may be interleaved across packets and is designed for high levels of data loss through transmission systems which may introduce error bursts. Whilst in extreme cases SDTI can create occasional errors, the MPEG-2 TS FEC is unsuitable for correcting SDTI errors due to the interleave length and this FEC will not exist if the packets carried are

only 188 bytes in length. Consequently, the SDTI-PF format includes an optional non-interleaved FEC to correct for any errors which may occur through the SDTI link.

Figures 2a to 2c illustrate MPEG-TS packets with different FEC capabilities.

It is permissible for SDTI-PF TS blocks to use all three services, User data, Retiming and RS-FEC as needed.

SDTI-CP TRANSMISSION PACKETS

Type value = '9xh'

Length = variable depending on the presence of User, Count and FEC components.

SDTI-CP Transmission packets adopt a packet structure based on 188 byte MPEG-2 Transport Stream packets and are managed in an identical manner.

UNIDIRECTIONAL INTERNET PROTOCOL PACKETS (UNI-IP)

Type value = 'Axh'

Length = variable up to 65535 bytes.

Because Uni-IP packets can have a length in excess of that capable of being handled by the RS-FEC, Uni-IP packets would not normally be able to use FEC. However, if the Uni-IP packets are constrained by the source device to be less than the limit set for correct FEC operation, then it may be used with success on only those packets which are below 254 bytes in length. Users should also be cautioned that some RS decoders rely on a fixed packet length in order to operate correctly in a pipelined operation. So Uni-IP packets with fluctuating packet lengths may cause decoder problems if the RS-FEC is active.

ATM Cells

Type value = 'Bxh'

Length = 53 bytes.

ATM cells have a fixed length of 53 bytes to which the User, Count and FEC may be added as needed.

Although illustrative embodiments of the invention have been described in detail herein with reference to the accompanying drawings, it is to be understood that the invention is not limited to these precise embodiments, and that various changes and modifications can be effected therein by one skilled in the art without departing from the scope and spirit of the invention as defined by the appended claims.

4. Brief Description of Drawings

The above and other objects, features and advantages of the invention will be apparent from the following detailed description of illustrative embodiments which is to be read in connection with the accompanying drawings, in which:

Figure 1 is a schematic diagram of the payload area of an SDTI line;

Figures 2a to c are schematic diagrams of MPEG 2 TS packets;

Figure 3 is a schematic diagram of an SDTI payload area divided into slots and having a format in accordance with an embodiment of the present invention;

Figure 4 is a schematic diagram of the payload header structure;

Figure 5 is a schematic diagram illustrating the correction of jitter in accordance with an embodiment of the present invention;

Figure 6 is a schematic block diagram of a system, in accordance with an embodiment of the invention for transmitting MPEG 2 TS packets over an SDTI system;

Figure 7 is a schematic block diagram of an example of the timing data inserter and of the timing corrector of the system of Figure 7;

Figures 8 and 9 are schematic block diagrams of examples of the counters of Figure 7;

Figure 10 is a schematic diagram of an SDTI variable block used in another embodiment of the invention;

Figure 11 is a schematic diagram of a TLD block used in the other embodiment of the invention;

Figure 12 is a schematic diagram showing a hierarchy of SDTI variable blocks and TLD blocks used in the other embodiment of the invention; and

Figure 13 is a schematic diagram of a TLD compound packet.

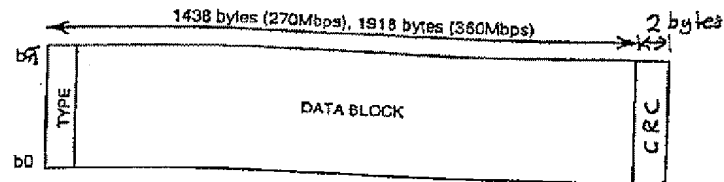


Figure 1: Format of the SDTI-TS Fixed Block

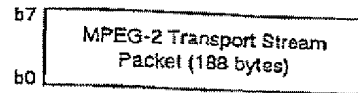


Figure 2a: MPEG-2 Transport Stream Packet

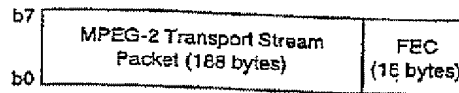


Figure 2b: MPEG-2 Transport Stream Packet with Active 16 byte FEC

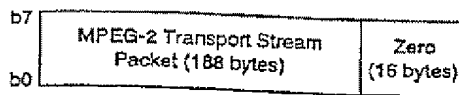


Figure 2c: MPEG-2 Transport Stream Packet with Null FEC Space

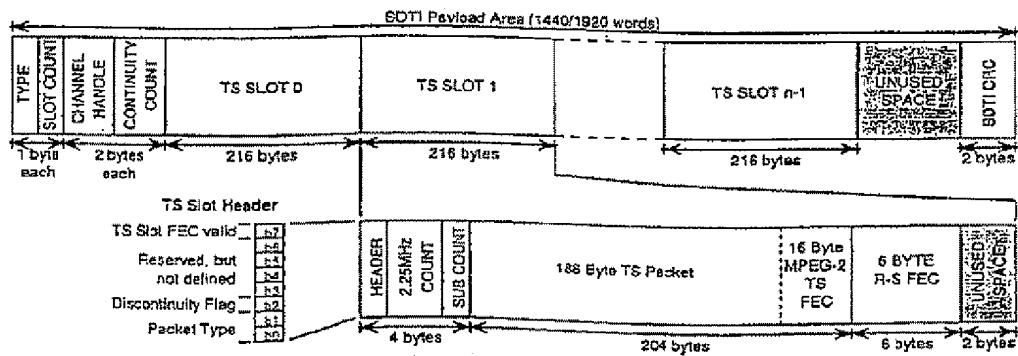


Figure 3: TS Payload Structure

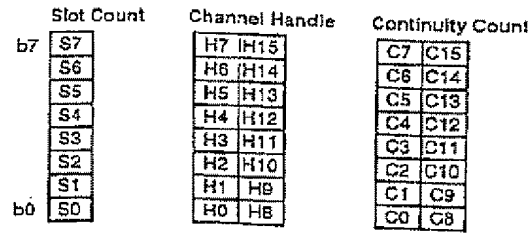


Figure 4: Payload Header Structure

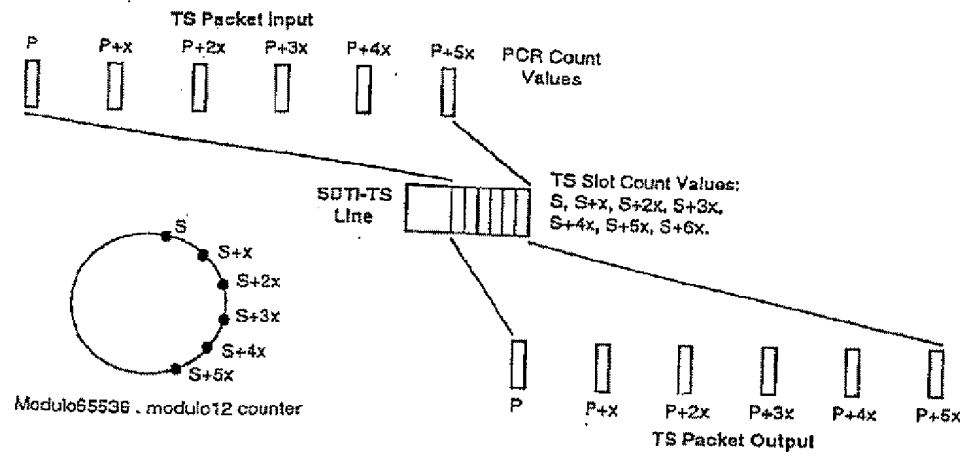


Figure 5: SDTI-TS Timing Reconstruction

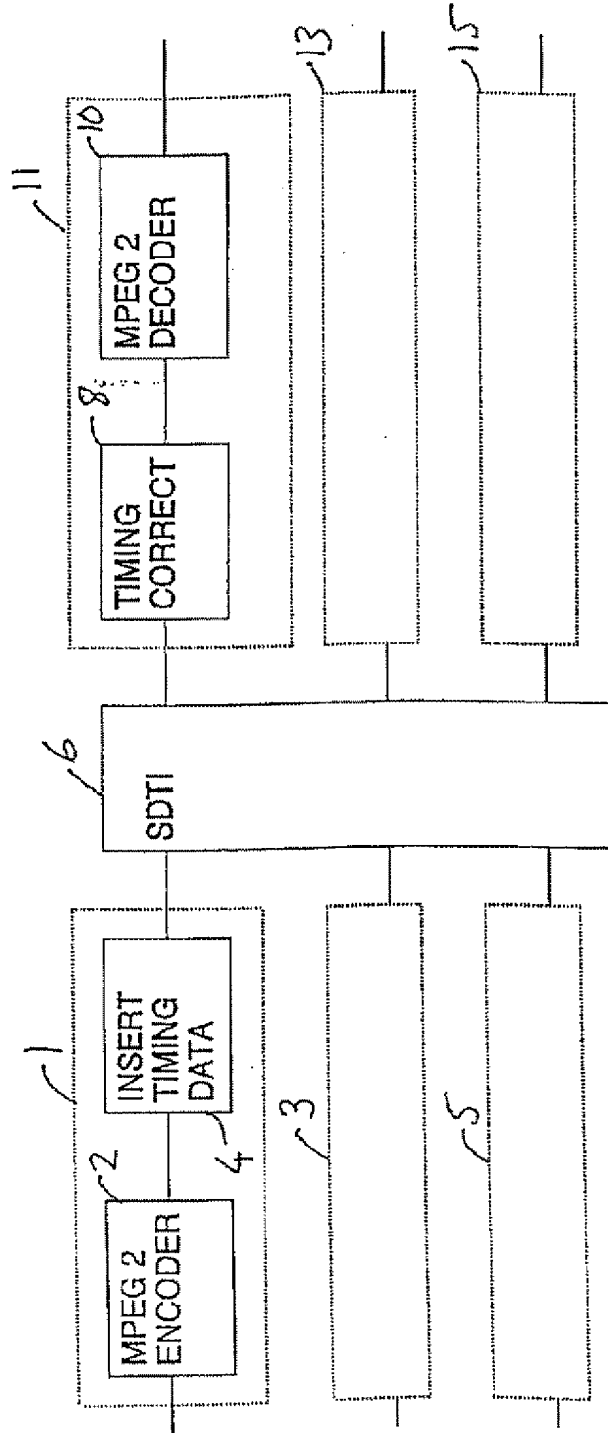


FIGURE 6



FIGURE 8

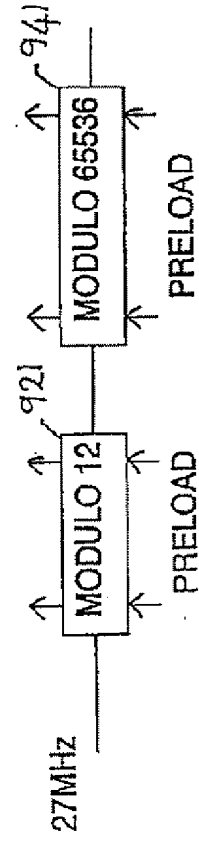
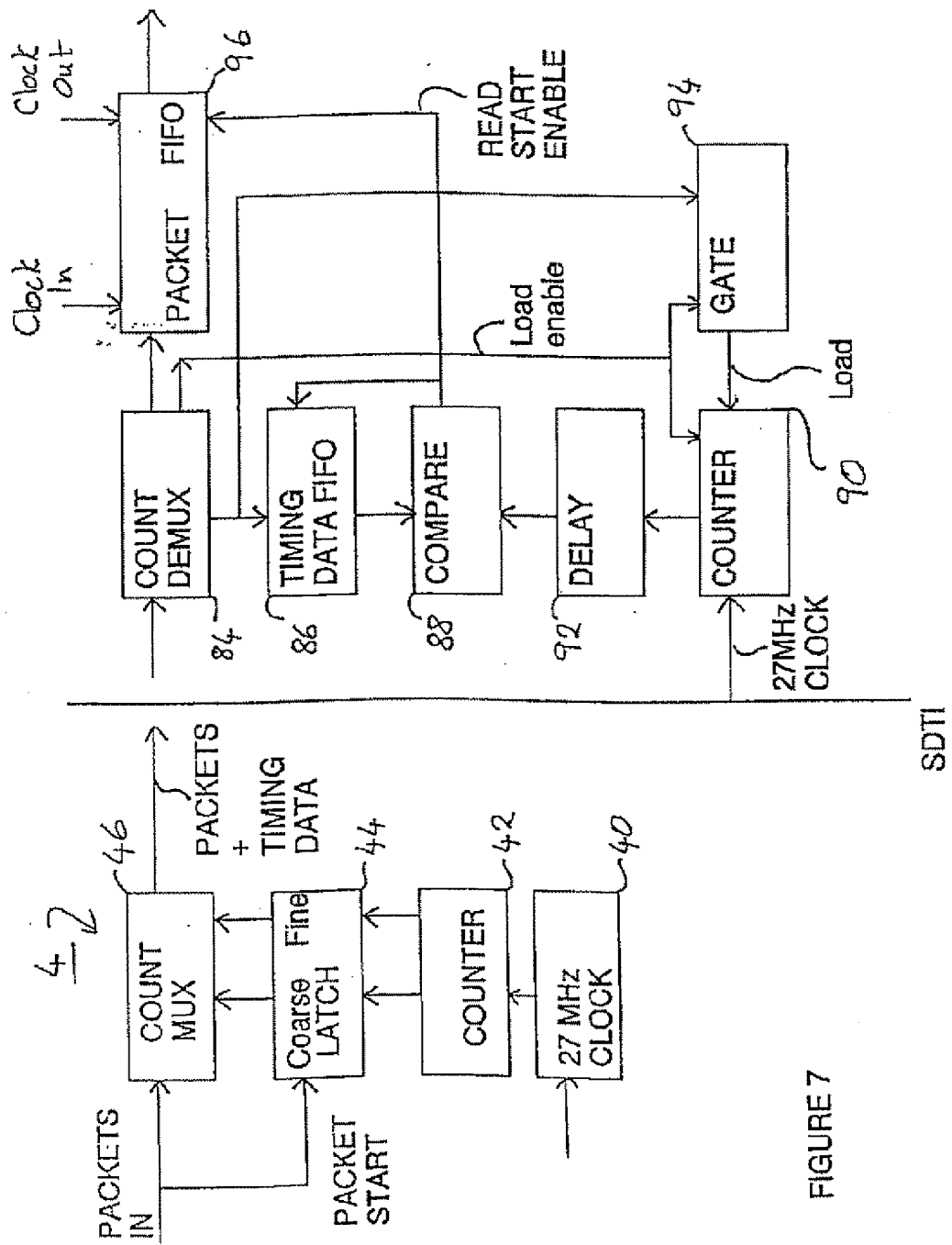


FIGURE 9



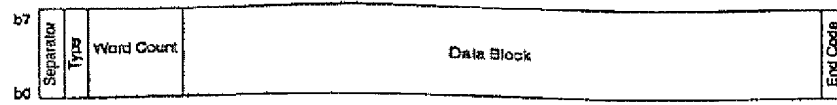


Figure 10 Format of the SDTI Variable Block



Figure 11 TLD Blocks

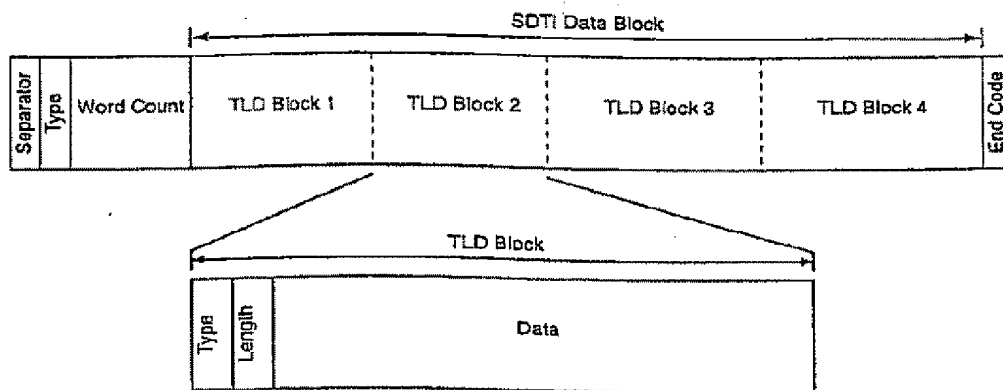


Figure 12 Hierarchy of SDTI Variable Blocks and TLD Blocks

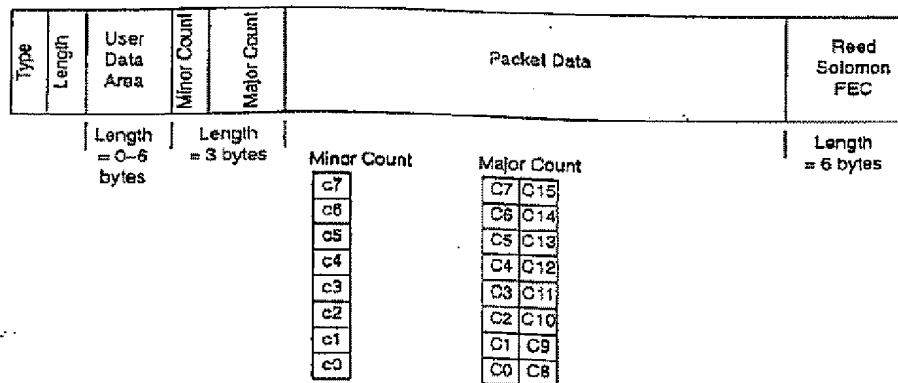


Figure 13 TLD Compound Packet With All Components

A digital signal comprises data blocks, each data block including a header containing data relating to the block and a plurality of slots. Each slot has a slot header relating to the slot and a data packet. The data packets contain successive parts of information from a source. A first slot contains a first packet containing a first part of the said information from the source also contains a reference time. The or each subsequent slot contains a subsequent packet of the information from the said source also timing information defining the timing of that packet relative to the reference time.

An encoder which encodes such a signal is provided. The corresponding decoder is enabled to correctly output the packets to allow correct decoding. Absolute delay of the packets has no effect on decoding. Jitter (i.e. variation in the timing of the packets relative to each other) may corrupt the decoding. The decoder compares the timing information of each packet with an internal clock set by the reference time of the first packet and outputs the packets when the clock time equals the packet time thus at least reducing the jitter.

The data blocks may be SDTI fixed length blocks and in a transmission system the packets are MPEG 2 TS packets, which are transmitted via an SDTI system. The data blocks may be SDTI variable length blocks and the packets may be MPEG2 TS packets, ATM cells or Internet Protocol packets.

2. Representative Drawing

Figure 3

PATENT ABSTRACTS OF JAPAN

(11)Publication number : 2000-174819

(43)Date of publication of application : 23.06.2000

(51)Int.Cl.

H04L 12/56

H04J 3/00

H04L 7/04

H04N 7/24

(21)Application number : 11-275487

(71)Applicant :

SONY UNITED KINGDOM LTD

(22)Date of filing : 29.09.1999

(72)Inventor :

WILKINSON JAMES HEDLEY

OLDAY STEPHEN CHARLES

(30)Priority

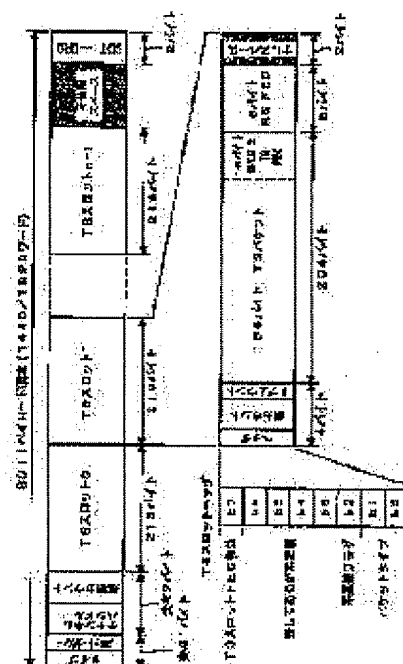
Priority number : 98 9821518 Priority date : 02.10.1998 Priority country : GB

(54) SIGNAL FORMAT, ENCODER, DECODER AND SIGNAL TRANSMISSION DEVICE

(57)Abstract:

PROBLEM TO BE SOLVED: To provide a flexible signal format and its signal transmission device.

SOLUTION: A digital signal is provided with data blocks, which includes a header including data concerning the blocks and plural slots. Each slot is provided with a slot header concerning the slot and a data packet. The data packet includes respectively the continuing parts of information from a source. The first slot includes a reference time, together with the first packet which includes the head part of the information. The succeeding slots include timing information with respect to the reference time of the packet together with the succeeding packets of the information. A jitter in the case of decoding is reduced by using an encoder/decoder for encoding and decoding the signal.



LEGAL STATUS

[Date of request for examination]

08.03.2006

[Date of sending the examiner's decision of rejection]

[Kind of final disposal of application other than the examiner's decision of rejection or application converted registration]

[Date of final disposal for application]

[Patent number]

[Date of registration]

[Number of appeal against examiner's decision of rejection]

[Date of requesting appeal against examiner's decision of rejection]

[Date of extinction of right]